

AmorFi: Amorphous WiFi Networks for High-density Deployments

Ramanujan K Sheshadri¹, Mustafa Y. Arslan², Karthikeyan Sundaresan²,
Sampath Rangarajan², and Dimitrios Koutsonikolas¹

¹University at Buffalo, SUNY

²NEC Laboratories America, Inc.

{ramanuja, dimitrio}@buffalo.edu

{marslan, karthiks, sampath}@nec-labs.com

ABSTRACT

The static capacity provisioning in traditional WiFi networks (WLANs) cannot cope with the high spatiotemporal traffic variations in high-density venues such as conference centers, stadiums etc. To guarantee reliable performance, venue owners are forced to over-provision their WLANs based on worst-case traffic demand estimations, increasing capital and operational expenses. We propose AmorFi, a radically new way of deploying WLANs to handle peak traffic demands with average-case provisioning. Our key idea is to decouple baseband processing from RF transmission (inspired by the cloud-RAN concept in cellular networks) and introduce software programmability to flexibly allocate WiFi capacity in real time based on varying traffic demands. We implement AmorFi using off-the-shelf WiFi APs over a RF-over-fiber cloud-RAN testbed. Our experiments and simulations demonstrate that the software-defined capacity allocation enabled with AmorFi delivers more than $2x$ throughput than traditional WLANs.

CCS Concepts

•Networks → Wireless local area networks;

Keywords

WLAN; Cloud RAN; Testbeds

1. INTRODUCTION

Despite the large WiFi deployment footprint and technological advances such as beamforming and MIMO, WiFi experience is still far from ideal in high-density venues such

as conference/event centers, stadiums, airports etc., where thousands of users demand reliable wireless service at once. The root cause of this problem is the disconnect between fixed WiFi provisioning (where the capacity is allocated a priori) and the high spatiotemporal traffic variation characteristically found in such venues.

State-of-the-art: Current WLANs in high-density venues are planned based on estimated per-user bandwidth requirements. Once an aggregate throughput requirement is determined for a particular area, the next step is to decide on the number of APs to provide the required capacity and assign channels to address inter-AP interference [15, 39]. The traffic demand is then distributed across the APs that serve the given area, using load balancing techniques.

When a traffic surge exceeds the planned capacity allocation, WiFi performance quickly deteriorates since it is virtually impossible to provide additional capacity in a reasonable time. To ensure reliable network-wide performance, venue owners are often forced to over-provision their WLANs based on worst-case traffic estimations, leading to increased equipment and maintenance costs [38]. In short, there is no efficient way for WLANs to adapt to changing traffic patterns other than manually deploying more APs (which is impractical) or worst-case provisioning (which is expensive).

In addition to the lack of adaptive capacity provisioning, today's WLANs also face the challenge of efficiently delivering broadcast traffic at a large scale. With the growing consumer interest in video applications (e.g., watching HD replays in stadiums), venues often feel the pressure to support wireless broadcast but do not have the mechanisms to do so efficiently [24, 27]. Traditional WLANs are known to exhibit many problems associated with broadcast traffic especially when it coexists with unicast applications such as e-mail and social media [10]. In this paper, we thus seek to answer the following question: *Can we satisfy worst-case traffic demands with average-case provisioning by allowing WLANs to change their capacity distribution in real time and support a heterogeneous mix of unicast (e.g., file download) and broadcast (e.g., live TV) applications?* To answer in the affirmative, we draw inspiration from a popular concept in cellular industry - cloud radio access networks or C-RANs.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

CoNEXT '16, December 12-15, 2016, Irvine, CA, USA

© 2016 ACM. ISBN 978-1-4503-4292-6/16/12...\$15.00

DOI: <http://dx.doi.org/10.1145/2999572.2999586>

A Potential Solution: The core idea in a C-RAN is to decouple radio frequency transmissions from baseband processing. This allows for multiple light-weight remote radio heads (RRHs) to be deployed for RF transmission, while the baseband processing is handled in a central processing cluster (i.e., a data center). The RRHs and the processing cluster is connected via a low-delay optical fiber network called the C-RAN front-haul (see Figure 1). The front-haul is the critical component that defines how the baseband processing units (BPUs) are mapped to RRHs, thus defining the capacity distribution across the network.

A given RRH supports simultaneous mappings from multiple BPUs. In a C-RAN-based WLAN, this would mean that multiple APs (acting as BPUs) can be mapped to the same RRH albeit on orthogonal channels. Note that since the C-RAN decouples an AP's capacity from its actual physical location, any AP can be mapped to any set of RRHs (not possible in a traditional WLAN). Thus, the C-RAN architecture allows dynamic re-purposing of capacity through appropriate re-configuration of the front-haul, thereby obviating the need for worst-case provisioning.

A Unique Benefit for WiFi: Unlike a cellular network, wherein each base station and user can communicate on multiple channels concurrently (i.e., carrier aggregation), this is not possible in WLANs. Even if latest WiFi clients support channel bonding up to 160 MHz in the 5 GHz band with 802.11ac APs, the legacy ones (11a/b/g/n clients) support 40 MHz bandwidth at most. Thus, legacy clients limit the effective bandwidth of the AP and create a performance bottleneck, warranting the deployment of as many APs as needed to use up all the orthogonal channels in the band (i.e., worst-case provisioning). C-RAN makes it possible to add radio resources (on different channels) to a RRH on the fly, allowing to borrow capacity from under-utilized areas of the network and use it to address the traffic surge where needed (i.e., avoiding over-provisioning and mitigating wastage).

Challenges: While adopting the C-RAN model in WLANs is a welcome step, the key challenge is orchestrating the C-RAN front-haul in real time to track and adapt to the spatiotemporal traffic fluctuations, so as to maximize the supported traffic demand. Further, with the increasing importance of wide-area broadcast applications (e.g., live sports replays [30]), the orchestration should consider unicast as well as broadcast traffic demands. While LTE has built-in support for wide-area broadcast [25], WiFi handles broadcast only at the cell level where each AP independently broadcasts the same content to its clients and contends for medium access (as it would for unicast traffic) [45]. If orchestrated properly, the front-haul has the potential to bring efficient wide-area broadcast support to WLANs. However, it is difficult to optimally satisfy both unicast and broadcast demands since each benefits from different front-haul configurations (detailed later in § 2).

AmorFi: We propose AmorFi ("Amorphous WiFi") – a first-of-its-kind C-RAN-based WLAN that adapts itself to cater to changing traffic demands (both unicast and broadcast). AmorFi realizes this through software-defined access (SDA), wherein it re-programs the C-RAN front-haul, to cre-

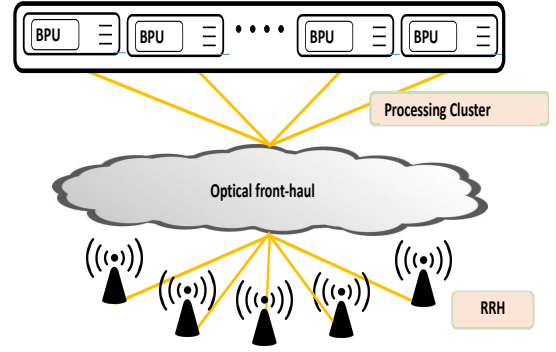


Figure 1: C-RAN architecture.

ate different cell configurations in real time. In designing AmorFi, we leverage the full potential of the C-RAN architecture for WiFi and make the following key contributions.

(1) *Designing algorithms to determine the optimal front-haul configurations.* Interference across WiFi APs (cells) is managed through CSMA and channel assignment. The latter is a hard problem in itself and requires taking into account interference conflicts between APs. The amorphous notion of cells in AmorFi makes this problem even more challenging. Further, we address an inherent tradeoff between wide-area coverage (useful for broadcast traffic efficiency) and spatial reuse (useful for unicast throughput) in determining the optimal configuration. AmorFi employs efficient algorithms to address this tradeoff and not only maximizes the traffic that can be supported, but also does so in an energy-efficient manner, employing a few BPUs.

(2) *Deploying AmorFi in practice.* This is challenging due to the lack of SDA support in today's WLANs. SDA requires AmorFi to re-map the signals on the fly from any AP (BPU) to any remote antenna (RRH), with negligible latency. We realize this in the analog domain using RF-over-fiber technology and use an optical switch to apply the desired front-haul configurations in real time. Using a testbed of four APs and four RRHs, we comprehensively evaluate AmorFi for both unicast and broadcast (video) applications. Our results indicate that AmorFi delivers twice the throughput supported by baseline schemes and supports a higher quality broadcast video. We also complement our testbed evaluations with large-scale simulations to demonstrate AmorFi's performance in larger deployments. A case study with IEEE Globecom 2003 attendance data reveals that AmorFi increases the median user throughput by more than 100%.

2. C-RAN BASED WLAN: AN OVERVIEW

2.1 Background

The C-RAN architecture (see Fig. 1) consists of the following components: **Processing Cluster:** centrally located baseband processing units (BPUs) responsible for signal processing and providing capacity. **Front-haul:** an optical fiber network that supports wavelength multiplexing to deliver the

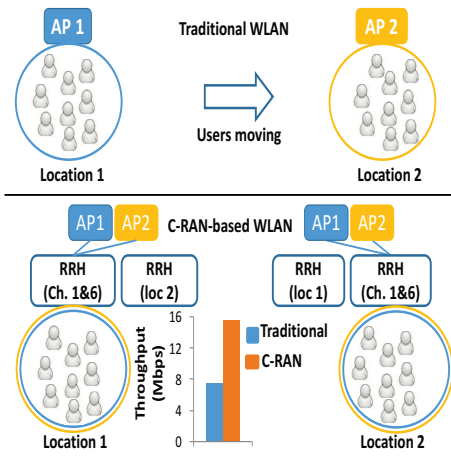


Figure 2: Repurposing capacity in C-RAN-based WLANs.

signals between the BPUs and remote antennas (RRHs), using either (a) CPRI (Common Public Radio Interface) - an interface standard for transmitting digitized IQ samples, or (b) RF-over-fiber, where analog RF is carried over the fiber. **Remote Radio Heads (RRHs):** light-weight antennas that convert optical signals to RF on the downlink (and RF to optical for uplink), providing wireless access to users. The front-haul is perhaps the most critical component that defines how the capacity of the BPUs is distributed across the RRHs. Next, we describe how a software programmable front-haul would help a C-RAN-based WLAN handle various scenarios that are challenging for traditional WLANs.

2.2 Architectural Advantages

We demonstrate the benefits of C-RAN with experiments conducted on our testbed (setup details deferred to § 4)

2.2.1 Repurposing Capacity

With user mobility, the traffic demand shifts across the network, potentially creating overloaded APs as well as leaving some APs under utilized. Past solutions that recommend centralized WLAN [8, 9], and other resource management techniques [47, 48, 5, 7, 34] are mostly intended at efficient load balancing and mitigating resource wastage during under-utilization. In the case of high traffic surge at a single location, even the most optimized algorithm of a traditional WLAN can only handle traffic demands equal to the capacity of the AP deployed at that location. Traffic demands higher than the capacity of the AP is dealt with either admission control (rejecting some users), or by simply providing sub-par performance to all (or few) clients. In contrast, the C-RAN based architecture offers the flexibility to repurpose the capacity of lightly loaded APs in overloaded areas, by converging the capacity of multiple under-utilized APs to a single location that requires additional capacity. To illustrate this with an example, we conduct an experiment (Fig. 2), where a group of 8 users move between two locations (location 1 and location 2) in our testbed. We keep the users close to the AP to avoid poor channel quality. The maxi-

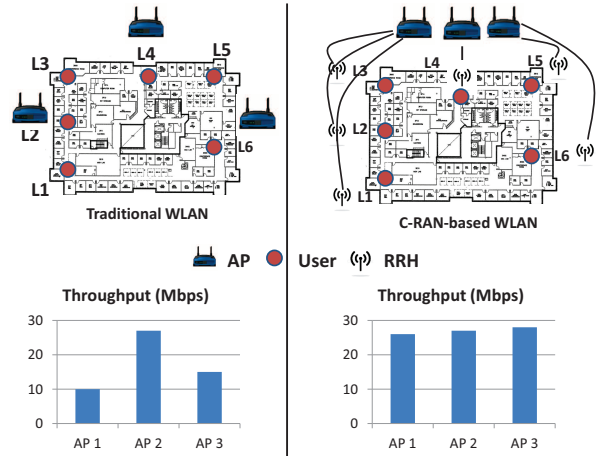


Figure 3: Larger AP coverage in C-RAN-based WLAN.

imum achievable per-user throughput in either location 1 or location 2 is measured to be around 60 Mbps (channel capacity). In the case of traditional WLANs, when all 8 users congregate at a single location (first in location 1 and then all move to location 2), they are all forced to share a single AP (one channel) - first, the AP at location 1 and then the AP at location 2. This results in a reduced per-user throughput of around 7.4 Mbps. In contrast, a C-RAN based WLAN gives the flexibility to map both the APs to the same RRH (on two different channels), enabling both the APs to serve at both the locations (initially at location 1 and then at location 2), as shown in Fig. 2. This allows the users to be re-distributed over two APs resulting in an increased per-user throughput of ≈ 15 Mbps), proving that software-defined front-haul is essential to gracefully handle the traffic surge.

2.2.2 Adaptable Coverage

In traditional WLANs, an AP is deployed in one physical location and its coverage area gets defined by its transmit power. Some users may be positioned far from an AP and thus have low data rates (e.g., in sparse traffic areas with low AP density). With packet-fair scheduling in WiFi, such low rate users adversely impact other users served by the AP. On the other hand, C-RAN-based WLANs allow mapping the signal of a given AP to more than one RRH simultaneously creating a one (BPU)-to-many (RRH) mapping in the front-haul. Since RRHs are light-weight (in cost and size), they can be deployed densely. With the help of one-to-many front-haul mapping, each user can then be served through its closest RRH with high data rates.

To understand the impact of coverage, we experiment with three APs and six clients. There are six candidate locations (L1-L6) to deploy the three APs (see Fig. 3). Once we choose a location for an AP, we let clients associate to their nearest AP and run multiple *iperf* sessions from each AP to its clients. We fix the locations that yield the best throughput for traditional WLAN (L2, L4 and L6). For C-RAN-based WLAN, we deploy RRHs at all six locations but use the same three APs to drive them (see Fig 3 for AP-

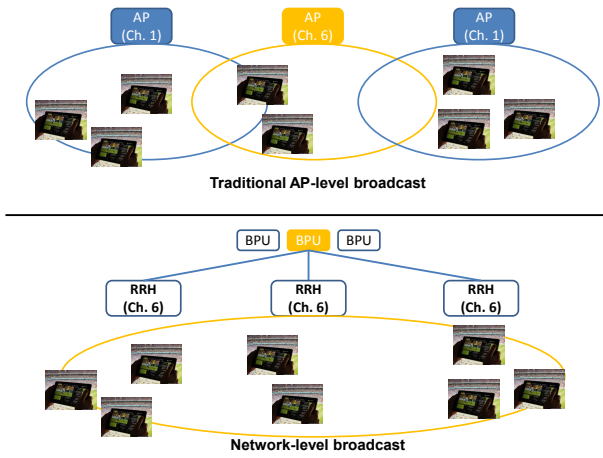


Figure 4: Network-level broadcast in C-RAN based WiFi.

RRH mapping). We see that per-user throughput is higher in C-RAN since each user receives data via its closest RRH (enabled by one-to-many mapping). Note that this is not possible in traditional WLANs due to limited AP transmit power. Even with the same power limit on each RRH, C-RAN-based WLAN allows for larger coverage owing to the decoupling of APs from physical locations and the transmit diversity benefit from synchronous RRH transmissions.

2.2.3 Efficient Network-level Synchronous Broadcast

In addition to unicast applications, supporting broadcast applications (e.g., live video) is also important in high-density venues [30]. Broadcasting MAC frames is inefficient in traditional WLANs due to the lack of acknowledgments and the use of low bitrates [10]. Even with the advances made by earlier works ([27, 10, 42]) and proposals by the IEEE 802.11aa amendment to 802.11 [1], broadcast transmission remains at the AP-level, requiring each AP to independently broadcast data to its clients (Fig.4). With AP-level broadcast, APs contend for medium access (similar to what they would do for unicast) even though they deliver the same content. On the other hand, one-to-many mapping in the C-RAN front-haul enables what we call *network-level broadcast* where an AP broadcasts content over a wide area (see Fig.4). This leads to better transmission efficiency for broadcast, freeing up the remaining APs to serve other applications. We note here that LTE already has built-in support for network-level broadcast, where base stations use a synchronization protocol to jointly transmit the broadcast content. Thus, the notion of one-to-many front-haul mappings is not as critical in cellular C-RANs as it is for C-RAN-based WLANs, where the latter benefits from synchronization in the optical domain overcoming the lack of an existing protocol.

2.3 Economic Viability

Plethora of studies [17, 20, 18, 19] argue that the C-RAN technology thrives in the form of micro C-RANs for high-density venues, and is economically viable considering the

long term operational cost benefits (up to 53%). With increasing number of enterprises gearing towards deploying micro C-RANs for cellular connectivity, there is a strong need for WiFi that can co-exist with and complement the cellular network. To this end, we believe AmorFi to be a perfect fit that can help create a C-RAN eco-system for all wireless connectivity. Since AmorFi can essentially be deployed on existing C-RAN infrastructure, the initial deployment cost can be minimal as well. Compared to traditional WLANs where low deployment costs are often offset by high operational costs [49, 36], a C-RAN based WLAN can indeed be economically beneficial for large enterprises, in the long run.

2.4 Determining Optimal Configurations

The true advantage of the C-RAN architecture in WLANs is SDA – orchestrating the front-haul (in software) to tailor capacity allocations based on prevailing traffic demands in real time. The spatiotemporal traffic variations requires constant tracking of the load and choosing the appropriate configuration. With dynamic front-haul configurations, the interference dependencies between APs change, making network planning a much harder task than it is for statically-deployed WLANs. Thus, even when the WLAN supports unicast traffic alone, it is challenging to come up with the optimal front-haul mapping that supports as much of the demand as possible.

When a WLAN supports a mix of unicast and broadcast applications, the problem becomes even harder since each application type has conflicting requirements on the type of front-haul mapping that needs to be adopted. With one-to-one mapping, each AP gets mapped to one RRH, creating smaller cells. This emphasizes spatial reuse, which increases the network capacity to support unicast applications. In contrast, one-to-many mapping creates larger cells (decreasing spatial reuse), which is important to efficiently deliver the same broadcast content over a wide-area. The problem is exacerbated by the clients not being able to seamlessly access multiple WiFi channels (unlike LTE), thereby requiring both unicast and broadcast applications to be delivered on the same channel (AP).

To summarize, the key challenge in C-RAN-based WLANs is to discover the optimal front-haul configuration that caters to both unicast and broadcast applications and adapt the configuration based on spatiotemporal changes in traffic demand. In doing so, one also has to manage interference conflicts across cells, which becomes harder since the cells themselves keep changing as determined by the front-haul.

Lack of Existing Solutions: To the best of our knowledge, no study in the traditional WLAN space addresses our problem of dynamic capacity provisioning to support unicast and broadcast traffic. There are some cellular C-RAN studies that advocate software-programmable front-haul such as [46], but they are not directly applicable to WiFi because of the following reasons. First, while WiFi channel bonding may aggregate capacity to some extent, it is limited due to the presence of legacy clients and CSMA requiring orthogonal channels between neighboring APs. Thus, it is essential

Table 1: Notations used in our problem

Symbol	Type	Description
\mathbb{A}	Input	The set of APs
\mathbb{R}	Input	The set of RRHs
\mathbb{K}	Input	The set of channels
T_r	Input	Traffic demand of RRH r
P	Input	Channel capacity of an AP (cluster)
\mathbb{N}	Output	The set of clusters
X_{ik}	Output	Binary variable denoting the assignment of channel k to cluster $i \in \mathbb{N}$
$Int(i_1, i_2)$	Output	Binary variable denoting that clusters i_1 and i_2 interfere with each other
$S(r)$	Output	The set clusters which include RRH r
$R(i)$	Output	The set of RRHs contained in cluster i

to map multiple APs to the same RRH to increase capacity and carefully assign channels to them to avoid co-channel interference. In contrast, cellular base stations and clients have access to all the channels in the spectrum; a base station can handle the traffic surge by increasing its bandwidth as needed (i.e., not needing channel assignment). Second, the lack of network-wide broadcast support in WiFi makes one-to-many front-haul mapping more critical than it is for cellular. Thus, one has to balance between supporting unicast and broadcast traffic in C-RAN-based WLANs since each requires different front-haul mappings.

3. AMORFI: DESIGN

In our network model, \mathbb{A} and \mathbb{R} denote the set of APs (BPU) and RRHs, respectively. While RRHs act as transmit/receive points, APs provide the wireless capacity. We envision AmorFi to track traffic fluctuations and adapt its front-haul configurations (AP-RRH mappings) at the granularity of epochs where measurements from previous epochs serve as input to drive the optimization for the current epoch. (The actual value of the epoch can be in minutes and tuned further based on the changing traffic patterns or past knowledge.) Hence, each RRH $r \in \mathbb{R}$ poses (requires) a traffic demand T_r for the current epoch (based on a weighted average from previous epochs), which is an estimate of aggregate throughput requirements for all users accessing the network via RRH r . The function $f : \mathbb{A} \rightarrow \mathbb{R}$ maps each AP to zero or more RRHs; the former denotes an idle AP (not serving users), while the latter defines a cluster i of RRHs (where $R(i) \in \mathbb{R}$), sharing the capacity P of the corresponding AP. P is estimated based on the aggregate throughputs delivered by the each of the RRHs in $R(i)$ from the previous epoch. When a cluster i is formed for an AP, it is assigned a channel $k \in \mathbb{K}$, where \mathbb{K} is the set of channels in the WiFi spectrum. $X_{ik} = 1$ captures this assignment (see Table 1).

Remarks: Although we do not explicitly consider external interference in our formulation, it must be noted that, interference estimation can be done using techniques described in [44, 14, 13], and the result can be easily factored in while determining the channel capacity of an AP (P).

3.1 Problem Definition

$ONC : \text{Maximize } \lambda$
 \mathbb{C}

such that:

$$(1) \lambda_r = \min\{1, \sum_{i \in S(r)} \sum_{k=1}^{|\mathbb{K}|} \frac{X_{ik} P_i}{T_i}\} \geq \lambda, \forall r \in \mathbb{R}$$

$$(2) \sum_{i=1}^{|\mathbb{N}|} \sum_{k=1}^{|\mathbb{K}|} X_{ik} \leq |\mathbb{A}|$$

$$(3) \sum_{i \in S(r)} \sum_{k=1}^{|\mathbb{K}|} X_{ik} \geq 1, \forall r \in \mathbb{R}$$

$$(4) X_{i_1 k} + X_{i_2 k} \leq 1, \forall (i_1, i_2) \in S(r), \forall r \in \mathbb{R}, \forall k \in \mathbb{K}, i_1 \neq i_2$$

$$(5) X_{i_1 k} + X_{i_2 k} \leq 1, \forall (i_1, i_2) \in \mathbb{N} \text{ s.t. } Int(i_1, i_2) = 1, \forall k \in \mathbb{K}$$

For an easier exposition, we describe the problem formulation and our solution for addressing unicast traffic first. Then, we extend them to accommodate broadcast traffic as well in Section 3.3. For a given number of APs (radios) and WiFi channels, AmorFi aims to maximize the amount of traffic demand (unicast) that can be satisfied at each of the RRHs in the current epoch. Towards capturing this objective, we formulate the following optimization problem, (ONC: optimal network configuration), where our goal is to find the network configuration (over the universe \mathbb{C} of all configurations) that maximizes $\lambda \in [0, 1]$. We define λ to be the minimum fraction of traffic demand satisfied across all the RRHs in the network. A network configuration includes the AP-RRH mapping (determining the capacity allocation) and the channel assignment to each of the clusters (APs).

The first constraint indicates that the fraction of traffic demand satisfied at each RRH r (denoted as λ_r ; called traffic satisfaction metric taking into account all APs serving it), is greater than λ . This makes λ the minimum such value (that is maximized) across the network. To calculate λ_r , we first determine the channel capacity share that RRH r has (out of the cluster capacity P), when it is included in a cluster together with other RRHs. Here, we assume that P is shared in proportion to the unicast traffic demand of each RRH in the cluster. The second constraint ensures that the number of clusters generated does not exceed the number of APs ($|\mathbb{A}|$). The third constraint ensures that all the RRHs (with users to serve) are covered in a cluster, and every RRH has an AP assigned. The fourth constraint requires that all the clusters sharing a common RRH, will be assigned different channels. The last constraint ensures that clusters that are in the interference range of each other, are not assigned the same channel.

Remark: It is important to map an AP to a *contiguous* set of RRHs, i.e., the clustered RRHs form a connected set, where connectivity captures RRHs that are within range of each other. The lack of such contiguity creates two drawbacks: (i) Mapping an AP to RRHs that are spread across the network, will reduce the spatial reuse for that channel.

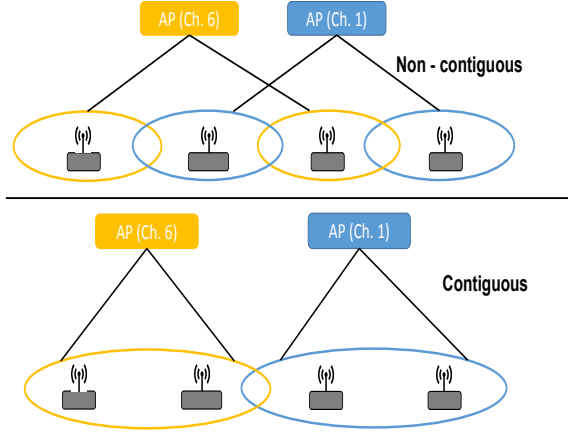


Figure 5: Non-contiguous vs. contiguous clustering.

(ii) As seen in Fig. 5, with non-contiguous clusters, a mobile user experiences frequent handoffs (not desirable for delay-sensitive applications). A contiguous cluster creates a virtual large cell that reduces such disruptions. We thus require that the clusters ($R(i)$) be contiguous.

3.2 Algorithms in AmorFi

3.2.1 Overview of Solution

ONC requires us to solve the problems of contiguous clustering and channel assignment *jointly*. Since each individual problem is NP-hard in itself, solving them jointly is much harder. Hence, we aim to design efficient, yet simple algorithms to solve ONC with the following intuition.

Intuition: Two parameters impact how much traffic can be supported: number of radios (APs) and channels. The number of APs control the maximum capacity that can be offered, and limit the number of RRH clusters that can be formed, while the clustering process itself determines how the available capacity is efficiently used (and not wasted) to satisfy the demand. When the channels are assigned to the clusters, they impact the supported traffic demand by a cluster, based on the sharing of its channel (and hence capacity) with other clusters in its neighborhood. While both clustering and channel assignment impact the net traffic served, their relative contribution to λ is biased, where clustering plays a bigger role. The impact of channel assignment varies with the number of channels. For example, it has little impact in 5 GHz, where many channels (nine) are available, making it easier for a conflict-free assignment, compared to 2.4 GHz (three channels), where it has appreciable impact.

Optimization: Given this biased effect, we see that the additional gains from solving the two problems jointly are not appreciable to justify the high complexity. Hence, AmorFi decouples the clustering process from channel assignment for a two-step approach as follows. It solves the clustering problem first, without accounting for channel assignment (conflicts) between clusters (APs), to maximize λ . Given the clustering solution, it then assigns channels to clusters to resolve as many of the inter-cluster conflicts and retain as

much of the λ (traffic satisfaction) from the clustering process ($\hat{\lambda}$, where $\hat{\lambda} \leq \lambda$). While this decoupled approach is less complex, it faces the challenge that the clustering solution that maximizes λ in the first step, may not contribute to a final solution (after channel assignment) with the best traffic satisfaction ($\hat{\lambda}$). In other words, a clustering solution with a smaller λ (after clustering) could potentially incur a less loss during channel assignment to yield a higher $\hat{\lambda}$.

To address this challenge, AmorFi adopts an iterative optimization approach. At the start of each epoch, the controller receives the aggregate traffic demand of each RRH (based on estimates from previous epochs). It then runs the iterative optimization, where it solves a dual of the problem in each iteration to determine the best network-wide configuration.

Step 1 - Initialize: Initialize λ with an upper bound (λ_0) on the traffic demand that can be satisfied at each RRH. This represents a configuration with maximum spatial reuse by mapping each AP to a separate RRH.

Step 2 - Cluster: In each iteration i , translate the λ -maximization problem into its dual; given a λ_i , determine the smallest number of APs (clusters) required to satisfy λ_i of the net traffic demand in each of the clusters. Here, impose a logical connectivity structure on the network of RRHs using breadth-first-search (BFS). The structure determines the smallest number of contiguous clusters of RRHs (m_i) that serve at least λ_i of the traffic demand in each cluster.

Step 3 - Assign Channels: Given the clusters and the available channels, try to find a conflict-free channel assignment. When the same channel has to be assigned to interfering clusters, update traffic satisfaction ($\hat{\lambda}_i$) to reflect the reduction in capacity (due to time sharing).

Step 4 - Iterate: Decrement λ by a small constant δ (say 0.01) for the next iteration ($\lambda_{i+1} = \lambda_i - \delta$) and repeat steps 2 and 3. Terminate when $\lambda = \delta$.

Step 5 - Output Configuration: Determine the network-wide configuration corresponding to the highest $\hat{\lambda}$, namely $\max_i \{\hat{\lambda}_i\}$ such that number of APs employed $m_i \leq |\mathbb{A}|$.

The time complexity of our solution is $O(I(|\mathbb{R}|^2 + |\mathbb{A}|^2))$, where I is the number of iterations, $O(|\mathbb{R}|^2)$ and $O(|\mathbb{A}|^2)$ are the complexities of the clustering and channel assignment components in each iteration, respectively. We now explain these steps in detail.

3.2.2 Initialize λ (Step 1)

At the start of an epoch t , the controller obtains traffic statistics from the previous epoch $t - 1$. Specifically, each AP a reports (i) average capacity P_a : the average of all the MCSs (Modulation and Coding Scheme) used for transmissions (from RRHs in its cluster) to all its clients; (ii) aggregate traffic demand $T^a(t - 1)$: aggregate of the traffic served to all its users per unit time of the epoch $t - 1$; and (iii) number of RRHs $n_a(t - 1)$ serving it. For epoch t , the controller combines AP reports to estimate: (i) P as the average of the capacity seen by all the active APs in the last epoch $t - 1$; and (ii) aggregate traffic demand per RRH r , $T_r(t)$ as the ex-

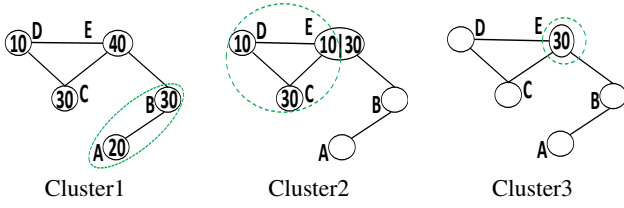


Figure 6: Clustering procedure (cluster capacity = 50 units)

ponentially weighted moving average of traffic demand from prior epochs, i.e. $T_r(t) = \alpha \cdot \sum_{a:r \in R(a)} \frac{T_a^a(t-1)}{n_a(t-1)} + (1 - \alpha) \cdot T_r(t-1)$, where $\alpha = 0.9$ is the filtering coefficient used in our implementation.

AmorFi starts by initializing λ to the theoretical upper-bound, calculated as the ratio of total network capacity to the total traffic demand. The total network capacity is the product of the number of APs ($|\mathbb{A}|$) and the average channel capacity (P), assuming that every AP gets a non-overlapping channel (upper-bound). Thus, $\lambda_0 = (P * |\mathbb{A}|) / (\sum_{r=1}^{|\mathbb{R}|} T_r)$.

3.2.3 Clustering (Step 2)

Since the traffic requirement at an RRH r (i.e. $\lambda_i \cdot T_r$, for iteration i) can be split and served by multiple clusters, the contiguous clustering problem to determine the minimum number of APs, can be optimally solved if we can construct a Hamiltonian cycle¹ on a graph where the RRHs are the vertices (edges denote interfering RRHs). Such a cycle would give us the ordering in which the RRHs should be traversed to produce the optimal clustering. However, even determining the existence of a Hamiltonian cycle is NP-complete in general graphs. Hence, AmorFi constructs an alternate ordering by using a breadth-first search (BFS) traversal on the RRH graph, rooted at the RRH with the least degree.

Algorithm: AmorFi clusters (groups) RRHs, level by level as shown in Fig. 6 (numbers in vertices represent unicast traffic demand), starting with the root (vertex A). Note that, a BFS ordering does not ensure that RRHs in the same level are connected (for contiguous clustering purposes). Hence, when AmorFi encounters an RRH that is not connected to the current cluster, it skips the RRH and moves to the next RRH (that is connected) in the ordering, either in the same level if available or next level otherwise. The current cluster terminates when the next vertex's (RRH's) demand uses up the capacity (50 units) – as seen for cluster 1 where it starts with vertex A (20 units) and ends with vertex B (30 units) when 50 units are filled up. When it is not possible to add a vertex to the current cluster and completely cover its demand, AmorFi covers the vertex partially (i.e., satisfying some of its demand). This is shown for cluster 2 where it starts with vertex C (30 units), continues to cover vertex D (10 units) and ends with partially covering vertex E for 10 units (out of 40). Next, a new cluster is started from an un-covered RRH (at the current level) that has the low-

est degree. The process continues until all RRHs are completely covered. It is possible for the demand at an RRH to be covered by multiple clusters (APs), e.g., for vertex E. This happens when the demand at the RRH is partially covered by a first cluster; the RRH remains un-covered, with its remaining demand being covered later by other cluster(s). The BFS traversal of the RRH graph has a time complexity of $O(|\mathbb{R}|^2)$.

Sub-optimality: It is possible for a cluster's capacity to be under-utilized at the edges of the RRH graph, when no other connected RRHs can use up the remaining capacity. This could result in more APs (than an optimal scheme) being required to support a given λ . However, by intelligently using BFS and skipping vertices that violate contiguity, as well as initializing a new cluster with a low-degree vertex, AmorFi keeps this sub-optimality low. Further, in a cluster, where there is remaining capacity, we distribute the remaining capacity among its member RRHs to increase the traffic satisfaction of that cluster. Hence, $\lambda_i^r \geq \lambda_i$, $\forall r$.

3.2.4 Channel Assignment (Step 3)

The traffic demand satisfied at each RRH λ_i^r from the clustering process is feasible, only if there exists a conflict-free channel assignment. As the clusters are formed, AmorFi also maintains a cluster graph (i.e., a graph with clusters as vertices) along with its adjacency matrix C_{uv} . $C_{uv} = 1$ if (a) Cluster u and Cluster v cover a common RRH or (b) If one or more RRHs in Cluster u , are in the interference range of one or more of the RRHs from Cluster v . As the vertices are added to the cluster graph (as clusters are formed), AmorFi keeps track of the largest clique² in the cluster graph. Note that finding the largest clique in a given graph (i.e., not iteratively tracked) is however a hard problem.

AmorFi employs the DSATUR [6] coloring algorithm for assigning channels on the cluster graph (any other coloring algorithm could be used as well). The general idea is to select the vertex (i.e., cluster) with the highest saturation degree (i.e., number of differently colored neighbors) and assign to it the least admissible color with $O(|\mathbb{A}|^2)$ complexity.

Impact on λ : If the clique number of the cluster graph is greater than the number of colors (channels) – $|\mathbb{A}|$ – one or more adjacent vertices (clusters) will be assigned the same color, decreasing the corresponding clusters' λ_i^r . The number of orthogonal channels in 5 GHz (nine) is larger than typical clique sizes observed in our cluster graphs (two to six). Hence, while AmorFi can eliminate the reduction in λ_r in 5 GHz, this may not be possible in 2.4 GHz, where there are only three channels. In this case, the coloring algorithm will assign the same channel to one or more adjacent clusters (say n) in a clique with size more than three, thereby reducing the per-cluster channel capacity for these clusters by a factor of $\frac{1}{n}$. Correspondingly, the λ_i^r (from the clustering step) of all the RRHs in these adjacent clusters that shared the same channel, will be scaled by $\frac{1}{n}$. The minimum traffic satisfied across all the RRHs at the end of the feasible channel assignment step is given by $\hat{\lambda}_i = \min_r(\lambda_i^r)$.

¹A path through all the vertices s.t. each vertex is traversed once.

²sub-graph where vertices are connected to each other.

3.2.5 Determining Efficient Configurations

As we decrement λ_i in each iteration (in **Step 4**), fewer APs would be required to satisfy it. This is because, with less traffic demand, the size of each cluster would increase as more RRHs can be served by a single AP. Ideally, to maximize λ , it would be sufficient to stop at an iteration, when the number of APs required just equals $|\mathbb{A}|$ (input). However, channel assignment breaks this monotonic trend – while utilizing all APs would support the maximum traffic demand λ_i , it also increases the number of clusters and potentially the clique number, consequently incurring a large reduction to λ_i during channel assignment (resulting in a final lower $\hat{\lambda}_i$). Thus, it is important to search over the entire space of λ (100 iterations with $\delta = 0.01$) to evaluate the relative impact of both the steps (clustering and channel assignment) and identify the best network configuration (**Step 5**).

Remarks: The total execution time of the algorithm depends on various factors like the number of RRHs, number of APs, and the value of δ considered. To get an estimate of the algorithm execution time, we measured the time taken to run our naive python implementation of the algorithm. We selected layouts of 50 popular venues across US [16], and calculated the ideal AP-deployment graph for each of the selected venues using the guidelines in [3]. For each of the AP-deployment graph, we considered the scenario where $|\mathbb{R}| = |\mathbb{A}|$, $\delta = 0.01$, $|\mathbb{K}| = 9$ (No. of non-overlapping channels in 5 GHz), and a random distribution of the unicast traffic demand. We then measured the total execution time of our program on a desktop computer running Ubuntu 14.04 OS, and equipped with an Intel-Xeon processor [29]. We verified that the total execution time in each case to be in the order of few seconds (average = 6.83 sec), which is significantly less than the epoch considered (in minutes). Furthermore, we would like to point out that a more optimal implementation of the algorithm using techniques like MapReduce [21] can reduce the execution time significantly. Additionally, AmorFi's algorithms can be run in a distributed manner for different subsets of APs, in parallel. This will further enhance the execution efficiency of the algorithm. Thus, we believe that the computational overhead to determine the network configuration during each epoch is negligible.

3.3 Unifying Unicast and Broadcast traffic

Tradeoff: A characteristic of high-density venues is the co-existence of unicast and broadcast applications. The key challenge in addressing such heterogeneous applications is to account for their contrasting nature. If a user browses the Web (unicast traffic), she would increase the traffic demand at an RRH. However, if she subscribes to an already broadcasting video stream, she would pose no additional demand at the RRH. Optimizing for unicast requires an increase in the capacity at the RRH, favoring configurations with small clusters that enable higher spatial reuse in the network. In contrast, optimizing for broadcast requires expanding the coverage of already available capacity to include multiple

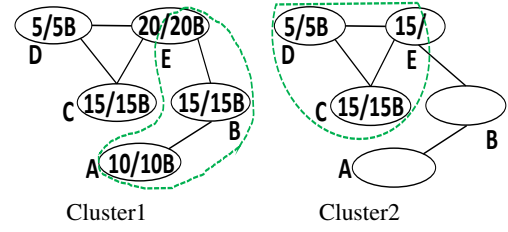


Figure 7: Clustering procedure - 50% of Beast traffic

RRHs, thereby favoring configurations with large clusters.

One way to address this is to optimize for unicast and broadcast traffic separately, e.g., by treating them as two individual problems. This would yield a solution where the APs would exclusively serve either unicast or broadcast traffic. In practice, a user requesting a broadcast stream while browsing the Web would have to be connected to two different APs (which is impractical), or suffer from poor performance in one of the applications. Thus, the key challenge is to identify a common network configuration that optimizes both the broadcast and unicast traffic jointly.

Problem Formulation: Given a separate traffic demand for unicast ($T_{r,u}$) and broadcast ($T_{r,b}$) traffic at an RRH, the ONC problem formulation remains the same except for the constraint on the unicast traffic satisfaction, which would get replaced by the following two constraints for broadcast and unicast respectively.

$$(1a) \lambda_{r,b} = \min\{1, \sum_{i \in S(r)} \sum_{k=1}^{|\mathbb{K}|} X_{ik} \gamma_i P_i\} \geq \lambda, \forall r \in \mathbb{R}$$

$$(1b) \lambda_{r,u} = \min\{1, \sum_{i \in S(r)} \sum_{k=1}^{|\mathbb{K}|} \frac{X_{ik} (1 - \gamma_i) P_i}{\sum_{l \in R(i)} T_l}\} \geq \lambda, \forall r \in \mathbb{R}$$

Here, we desire the same fraction of traffic satisfaction for both broadcast ($\lambda_{r,b}$) and unicast ($\lambda_{r,u}$) traffic at an RRH. γ_i is the output parameter that determines the amount of cluster (AP) capacity that is split between broadcast and unicast demands, and is responsible for the coupling between the heterogeneous demands.

Algorithm: Interestingly, our clustering algorithm is inherently equipped to address unicast and broadcast demand jointly, requiring only a minor tweak. Recall that AmorFi transforms the λ -maximization problem to an iterative minimization of the number of APs required to realize a given λ . Hence, at each iteration, it increases the size (coverage) of the clusters as much as possible, so as to minimize the number of APs employed. This transformation naturally lends itself to optimizing for broadcast traffic. Note that the most efficient use of a cluster's (AP's) capacity (say p) is to leverage the broadcast advantage and serve an equivalent amount (p) of broadcast traffic demand from as many RRHs as possible. Hence, given separate traffic demands for unicast ($T_{r,u}$) and broadcast ($T_{r,b}$) at an RRH and λ_i , AmorFi satisfies the broadcast demand ($\lambda_i T_{r,b}$) first, before satisfying the unicast demand ($\lambda_i T_{r,u}$) at the RRH. This allows AmorFi to leverage the broadcast traffic that is already

served by the cluster to also satisfy (completely or partially) the broadcast demand at the new un-clustered RRH being considered, thereby utilizing the cluster's remaining capacity to accommodate additional unicast/broadcast traffic. In other words, whenever a new RRH, \hat{r} , is added to a cluster c , only its additional broadcast demand (if positive) not satisfied by the cluster's existing broadcast traffic, i.e., $\lambda_i \cdot (T_{\hat{r},b} - \max_{r \in R(c)} \{T_{r,b}\})$, would eat into the cluster's remaining capacity as new traffic demand. Thus, the broadcast traffic satisfied by the cluster would increase to $\lambda_i T_{\hat{r},b}$. The unicast demand at the RRH is then handled as before.

Fig. 7 illustrates this algorithm where 50% of the total demand at each RRH is broadcast (B). The procedure starts at the root node (A) and cluster capacity is 50 units as before. When adding vertex B to cluster 1, the additional load from broadcast traffic is only 5 units since the cluster already serves 10 units of broadcast from vertex A. With vertices A and B in the cluster, 25 units of the capacity are used for unicast and 15 units for broadcast for a total of 40 units (leaving room for 10 units). Next, the cluster exhausts its capacity by completely covering the broadcast demand of vertex E (for additional $20 - 15 = 5$ units) and partially covering the unicast demand for 5 units. Cluster 2 initiates from vertex C and is formed similarly.

Remarks: AmorFi's transformed optimization determines efficient network configurations that balance the coverage benefits of large clusters for broadcast and the reuse benefits of small clusters for unicast. If the traffic demand is biased towards a particular traffic type, AmorFi will automatically tailor the configurations to favor the dominant traffic. Further, AmorFi's approach of delivering a given traffic demand using less radios, also makes it energy-efficient.

4. IMPLEMENTATION

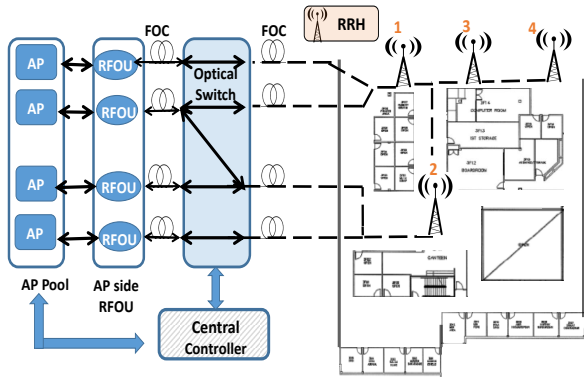


Figure 8: Our system architecture and testbed layout

Our system architecture is shown in Fig. 8. The APs are laptops with dual-band 802.11 a/b/g/n WiFi adapters. The main intelligence of AmorFi resides in the central controller, which is a PC that collects AP reports and executes our algorithms. It then instructs the APs to apply the channel assignment decisions and configures the switching unit to effect the

AP-RRH mappings. The mappings are realized in the optical domain where a programmable switching unit (i.e., an optical splitter) maps each AP to zero or more RRHs, allowing a mix of one-to-one and one-to-many configurations [23]. Since the switching is entirely in the optical domain, the switching latency does not incur any excessive overhead. Modern optical switches have switching latencies of a few tens of nanoseconds [35]. We have four APs and four RRHs due to limited number of ports on our optical switch. The RF-to-optical units (RFOUs) are transceivers that convert the RF signal to optical for the downlink, and vice-versa for the uplink [37]. We remove the external antenna of an AP and attach it to the RFOU to provide the RF source at the AP side. On the RRH side, RFOUs transmit/receive RF signals providing wireless access to clients. The RRHs in our testbed have inbuilt power moderator which provides a constant TX power of $\approx 17dBm$ irrespective of the input optical signal quality. Currently, our RRHs can be fitted with only one RF antenna, restricting us to have SISO experiments. However, we also create a proof-of-concept MIMO deployment as detailed in Section 5. The front-haul is based on Radio-over-fiber (RoF), where an optical signal is modulated based on the input radio signal, and transmitted over Fiber Optic Cable (We used a single mode fiber with 1550 nm wavelength in our testbed) [40]. Optical multiplexing helps us carry multiple optical signals (corresponding to multiple RF signals) on the same fiber using different wavelength, allowing RRHs to provide simultaneous service from multiple APs (on different WiFi channels). With fiber latency around $4 \mu sec/Km$, RoF retains the signal synchronization across RRHs, as well as the timing constraints between uplink and downlink signals. The RoF latency when compared to the minimum DIFS duration in 802.11 ($34 \mu sec$, and $50 \mu sec$ in 5 GHz, and 2.4 GHz band respectively) is almost negligible, allowing APs to carrier sense at the RRHs without any significant delay. Apart from RoF's simplicity, it also has low signal attenuation over long distances when deployed in large venues. Note that a C-RAN-based WLAN can be realized based on technologies other than RoF, e.g., CPRI. AmorFi is compatible with such architectures as well since it does not rely on RoF being available.

5. PERFORMANCE EVALUATION

5.1 Prototype Evaluation

We first evaluate AmorFi with only unicast traffic and later conduct experiments to include broadcast traffic as well. Our RRHs have interference relationships as in Fig. 9, where an edge indicates that the two RRHs overhear each other.

Baselines: We compare AmorFi against two baselines: (a) Traditional WLAN, with no load balancing (Tr_no_LB) - agnostic to AP load, where the clients associate to the AP with the strongest signal and (b) Traditional WLAN with load balancing (Tr_LB) - distributes the clients evenly among all APs that can potentially serve the clients (possibly with weaker signal), to avoid unfair load distribution among APs. We implement these baselines by static one-to-one config-

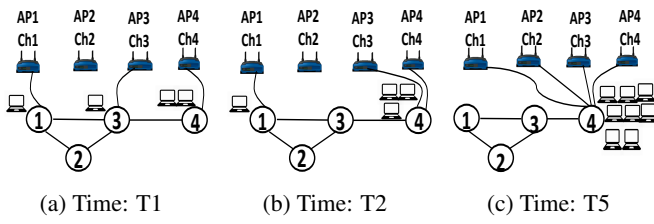
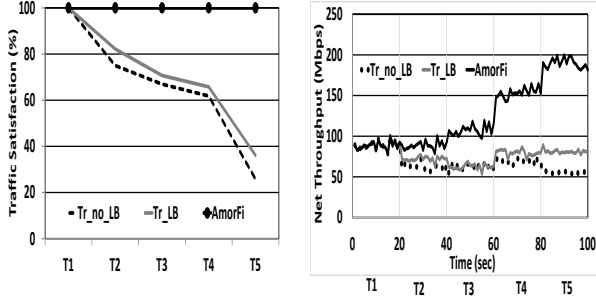


Figure 9: Scenario I : AmorFi 's AP-to-RRH configuration



(a) Scenario I: Traffic satisfaction
(b) Scenario I: Throughput

Figure 10: AmorFi Vs. baselines during traffic surge.

uration between the APs and RRHs (AP1 to RRH1, AP2 to RRH2 and so on). We conduct our experiments at night without any external interference in the 5 GHz band. Each AP is assigned an orthogonal 20 MHz channel. We represent these channels as Ch1, Ch2, Ch3 and Ch4.

We used *iperf* to measure the maximum per-AP throughput for UDP and TCP to be ~ 58 Mbps and ~ 49 Mbps, respectively with 64-QAM (one spatial stream, 800 ns. guard interval and 65 Mbps bit rate). The total UDP and TCP network capacity with four APs is thus ~ 232 Mbps and ~ 198 Mbps, respectively. We used eight Wi-Fi clients and set the traffic demand of each client to $\frac{\text{Network capacity}}{8}$, meaning that an AP can only satisfy the traffic demands of two clients.

Scenario I: Handling Traffic Surge: We first test the case when traffic demand gradually increases due to new clients arriving over five time intervals (**T1-T5**), each interval lasting 20 seconds. The initial AmorFi mapping is similar to the baselines where each AP is mapped to a separate RRH. Fig. 9 shows how this configuration evolves to accommodate the increase in load at the end of a subset of the intervals. In Fig. 10, we show the traffic satisfaction metric ($\lambda = 1$ is represented as 100%) and the TCP throughput for the three schemes. UDP results are similar and thus omitted. Initially, all schemes satisfy 100% of the demand since each AP serves at most two users. During **T2**, one user moves from RRH3 to RRH4 as in Fig. 9b. Without load balancing, traditional WLAN connects this user to AP4 (mapped to RRH4), overloading the AP since it ends up serving three users. With load balancing, the same user associates to AP3 (via RRH3), avoiding overload at the expense of reduced throughput due to lower SNR. With the ability to map multiple APs to an RRH, AmorFi re-maps AP3 (from RRH3) to

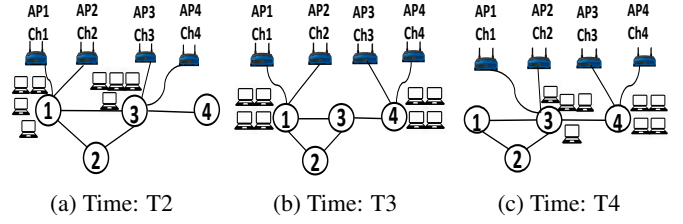


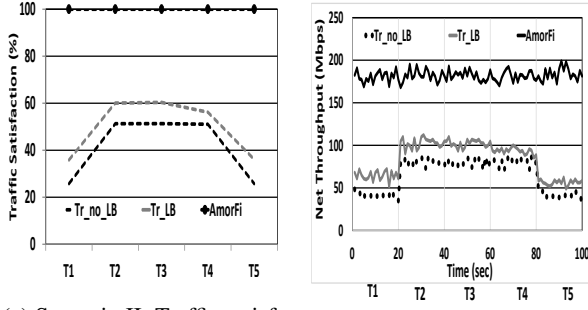
Figure 11: Scenario II : AmorFi 's AP-to-RRH configuration

RRH4 in addition to AP4 (see Fig. 9b). AmorFi also balances the load at RRH4 between AP3 and AP4, avoiding overload without the SNR penalty for the new user. Such ability to re-purpose the capacity of under-utilized APs on the fly is unique to a C-RAN-based deployment and is one of the key features enabled by AmorFi .

In subsequent intervals (**T3, T4**), as new clients join, the baseline schemes have lower traffic satisfaction with load balancing being slightly better (66% vs 61%). The final time interval **T5** shows the case when all eight users congregate at RRH4. AmorFi maps all four APs to RRH4 on different channels (Fig. 9c), distributing the clients evenly and efficiently utilizing the capacity. This helps AmorFi completely satisfy the demand while the baseline schemes satisfy 38% (with load balancing) and 25% (no load balancing) of the demand. The higher traffic satisfaction with AmorFi also reflects on aggregate throughput, where it delivers more than twice the throughput of baseline schemes as seen in Fig. 10b.

Scenario II: Handling Traffic Migration: We now test the case when the traffic demand is constant, but shifts across the network due to user mobility. The configurations of AmorFi and the comparisons with baseline schemes are shown in Figs. 11 and 12. We demonstrate AmorFi 's ability to move capacity end-to-end in a network to address the spatial movement of traffic demand. During **T1**, eight users are concentrated at RRH1 and AmorFi maps all four APs to RRH1. During **T2** through **T4**, users in groups of four move towards RRH4 as seen in Fig. 11, and all users end up at RRH4 during **T5**. While the baseline schemes achieve traffic satisfaction of only 60% and 51% in the best case (**T2-T4**), and only 36% and 25%, respectively in the worst case (**T1** and **T5**), AmorFi always maintains 100% traffic satisfaction by distributing the demand evenly between the APs , which also increases the aggregate network throughput (Fig 12b).

Scenario III: Network-level Video Broadcast with Background Unicast traffic: Broadcasting videos over WiFi is becoming increasingly popular in high-density venues. Accommodating such high-bandwidth broadcast applications is challenging since any increase in the unicast traffic demand will invariably affect the quality of the broadcast video. We conduct an experiment where we measure the video quality of broadcast videos while varying the background (unicast) traffic. We use three 8 Mbps video streams as broadcast traffic (delivered by VLC). The total broadcast traffic constitutes about 40% of the channel capacity of an AP. We modified the original WiFi driver to transmit broadcast MAC frames at higher bit rates (the stock driver allows only basic rates).



(a) Scenario II: Traffic satisfaction (b) Scenario II: Throughput

Figure 12: AmorFi Vs. baselines during traffic migration.

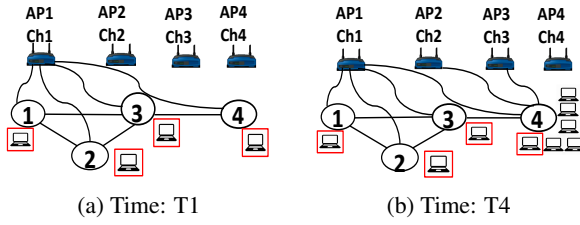


Figure 13: Scenario III: AmorFi's AP-to-RRH configuration

We fix the MCS of all the APs to 64-QAM and keep the clients in close proximity to the RRHs.

Experiment: The experiment spans four one minute long time intervals (**T1-T4**). Fig 13 shows the initial user distribution during **T1**. Users subscribing to broadcast traffic at each RRH, are collectively represented by one user with a box around it. To serve the broadcast demand, the baseline schemes require all four APs (and four channels) to broadcast the same videos at cell level. Owing to its network-wide broadcast capability, AmorFi requires only one AP (and one channel) mapped to all four RRHs to deliver broadcast videos to all subscribers (see Fig. 13a). Starting with **T2**, unicast users (each with traffic demand of 20 Mbps) gradually join the network at RRH4. As the unicast traffic demand increases, video quality with baseline schemes suffer significantly as indicated by the drop in PSNR (in Fig. 14) of the videos received by a user at RRH4. On the other hand, AmorFi maps additional APs (on channels 2 and 3) to RRH4 and utilizes them to serve the unicast users. This allows the AP on channel 1 to continue serving the broadcast traffic over four RRHs (Fig. 13b), yielding a much better video than the baseline schemes.

5.2 Mobility Support in AmorFi

Even though we do not explicitly optimize for mobility, one-to-many configurations (large cells) in AmorFi aids in reducing the disruptions due to handoffs for mobile users. We verify this by moving a user between two RRHs connected to the same AP (creating a large virtual cell). We compare the TCP throughput to a traditional WLAN where the two RRHs are mapped to two different APs, requiring the user to handoff from one AP to the other. As seen in

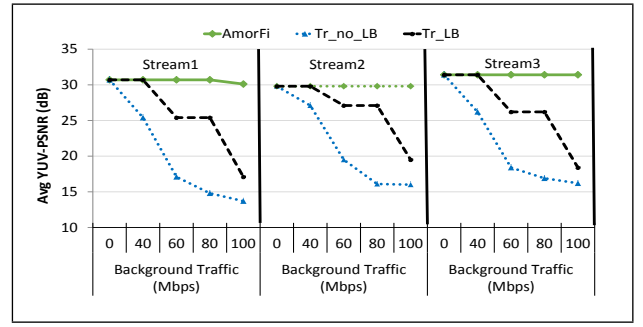


Figure 14: PSNR of received broadcast video

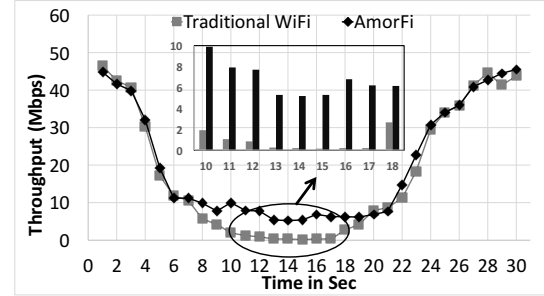


Figure 15: Mobility benefits with AmorFi

Fig. 15, AmorFi results in a much more graceful transition as the user leaves the first RRH and enters the coverage of the second RRH.

5.3 Realizing MIMO in AmorFi

MIMO configurations can be realized just as easily as SISO using RRHs with multiple antennas. We enable 2x2 MIMO in our testbed by placing two RRHs next to each other to emulate a “super RRH” with two antennas. Using the four RRHs, we thus have two super RRHs in two locations. Each antenna port of a MIMO AP (each corresponding to a spatial stream) is split (optically) and fed to one of the RRHs in each super RRH location. This creates a MIMO system with transmit diversity since a spatial stream

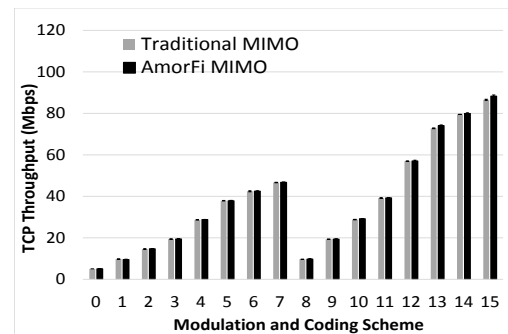


Figure 16: AmorFi MIMO: TCP Throughput

is served by two RRHs at different locations. We place a user in between our super RRHs and compare its TCP throughput to a case where it is placed in close proximity to a MIMO AP (i.e., no C-RAN involved). As seen in Fig.16, the throughputs are highly similar suggesting that our testbed successfully emulates MIMO transmissions in AmorFi.

5.4 Large Scale Simulations

To evaluate AmorFi in larger deployments, we consider the layout of 50 popular venues across the US [16]. Due to lack of any publicly available data (to the best of our knowledge) on the WLANs in these venues, we followed the guidelines in [3] to build the best possible representation of the ideal AP deployment in each venue. For each of the 50 venues, we created WLANs that represent low, medium, and high AP density for a total of 150 deployments.

The peak hour traffic demand is estimated to be equal to the total capacity of a deployment, calculated as the product of the number of APs ($|\mathcal{A}|$), and channel capacity P . We consider three different load distributions: (a) 60-40: 60% of the total traffic demand is randomly distributed across 40% of the RRHs, (b) 70-30 and (c) 80-20. We measure the traffic satisfaction metric (λ), using a custom simulator when (a) number of APs = 25% of RRHs (b) number of APs = 50% of RRHs (c) number of APs = 100% of RRHs. To have a fair comparison, especially for the first two cases when the number of APs are fewer than the number of RRHs, we create a new baseline scheme that also has C-RAN front-haul capability albeit with simpler clustering than AmorFi

Baseline: The baseline decides on the AP-to-RRH mapping by spatially partitioning adjacent RRHs into clusters, such that the number of clusters formed is equal to the number of APs. The number of RRHs per cluster is given by $\frac{\#of\ RRHs}{\#of\ APs}$. When the number of APs are equal to the number of RRHs, it converges to a traditional WLAN, where each AP is mapped to a separate RRH.

5.4.1 Results

In the interest of space, we only discuss the results when the number of APs is 50% of the number of RRHs. However, our observations and conclusions remain the same for other cases.

Fig 17a, 17b, 17c show the CDF of the traffic satisfaction ratio for different load distributions, considering only unicast traffic. AmorFi-2.4 represents AmorFi in 2.4 GHz (three orthogonal channels) and AmorFi-5 represents AmorFi in 5 GHz (nine channels). We see that AmorFi-2.4 and AmorFi-5 increase the median traffic satisfaction (by 33 percentage points and 75 percentage points, respectively) when averaged over the three load distribution cases. AmorFi performs worse in the 2.4 GHz band due to fewer channels than the 5 GHz band. Thus, when our clustering algorithm reveals a cluster graph with a large clique size (e.g., five), the reduction in λ in the channel assignment stage is correspondingly higher (since some neighbor clusters are assigned the same channel). In 2.4 GHz, we observe that the configuration with the maximum λ is usually the one with smaller clique sizes,

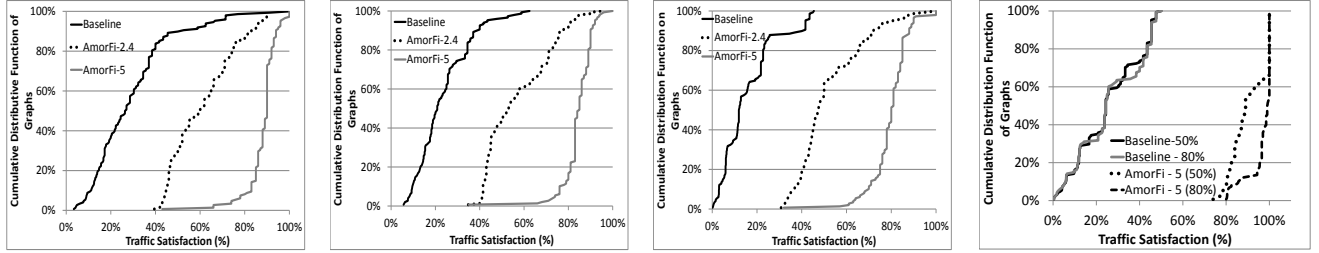
utilizing a smaller fraction of the available APs compared to the 5 GHz band. 5 GHz band allows more APs to be used in highly-loaded areas (i.e., larger cliques) without significant co-channel assignment penalty.

We also observe that a given scheme performs worse for higher skew in load distribution (e.g., with 80-20 case). With such high skew, few RRHs have very high load requiring many APs mapped to them. However, the number of channels restrict how many APs can be effectively mapped to an RRH making it more difficult to support the demand in highly skewed regimes. Nevertheless, we observe that AmorFi provides 80% median traffic satisfaction even with the 80-20 case owing to its efficient clustering algorithm. Interestingly, we observe that the baseline scheme performs the same in both bands. Upon closer inspection, we find that the spatial division of RRHs into clusters creates a cluster graph for the baseline where the clique size is at most three. Thus, the baseline cannot take advantage of the additional number of channels available in the 5 GHz band.

Finally, Fig. 17d shows traffic satisfaction ratio (for 80-20 load distribution), when broadcast traffic comprises 50% and 80% of the demand at each RRH. We see that the baseline scheme performs similarly in both cases. This is because the static partitioning of RRHs into clusters is agnostic to the traffic distribution within each cluster. Since AmorFi explicitly accounts for broadcast and unicast traffic demands and has the front-haul support for mixed configurations, it provides a traffic satisfaction of more than 90% while the baseline provides only about 25%.

5.5 Case Study: IEEE Globecom

Finally, we test AmorFi using NS3 simulator, based on the attendance data of a real event – IEEE Globecom 2003, which was held at the Marriott hotel in San Francisco with 1447 people attending the event [26, 33]. We consider the conference events on 2nd and 3rd of December, when the conference scheduled multiple technical presentations between 9AM-12PM and 2PM-5PM. There was also a luncheon on 2nd between 12PM-2PM, and a banquet on 3rd between 6PM-9PM. Technical presentations were held in rooms which was 77 sq. meters, and the social events was in a ballroom which was 3681 sq. meters. Based on this data, we deduced the AP interference graph using [3]. To make the results relevant to the present day, we assume all the APs and clients to implement the 802.11n WiFi standard, and thus emulate modern WiFi network. With traditional WLAN, the APs at the ballroom would be under-used during technical presentations, while the APs in the technical presentation rooms would remain idle during the social events. Assuming the attendees to be equally distributed in each event, we calculate the best throughput each user can get using the NS3 simulator based on the AP locations and the seating arrangements [33]. This throughput estimate is then input as traffic demand to AmorFi. Fig 18 plots the CDF of per-user TCP throughput using the traditional WLAN (with load balancing -Tr_LB) and AmorFi, when 50% of attendees accessed the network. We see that AmorFi clearly outperforms traditional WLAN by providing more than $2x$ median through-



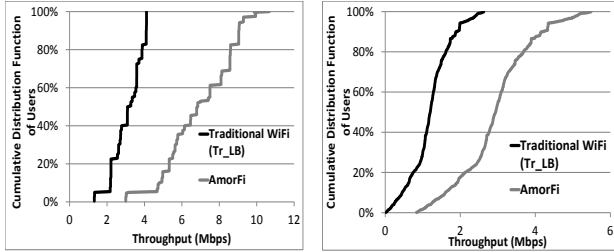
(a) 60-40: Unicast traffic

(b) 70-30: Unicast traffic

(c) 80-20: Unicast traffic

(d) Broadcast -5Ghz

Figure 17: CDF of Traffic satisfaction - No. of APs = 50% RRHs



(a) Technical Sessions

(b) Social Gathering

Figure 18: Simulated per-user throughput in Globecom 2003

put. Interestingly, the absolute performance is less but the gain (over the baseline) is more for the social events ($2.5x$ vs. $2.2x$) since they have a higher user concentration (similar to the highly skewed load distribution in our simulations).

6. RELATED WORK

Traditional WLANs: Supporting WLAN traffic demand is a problem that has been extensively studied both in academia and industry (e.g., [47, 48, 5, 7, 4, 15, 2, 34]). The proposed solutions mainly incorporate improved channel management and load balancing, which are some of the functionalities that we also implement in AmorFi. However, our objective is not to come up with a better channel assignment or load balancing method (AmorFi can incorporate existing procedures). Instead, AmorFi proposes a novel architecture for WLANs and addresses unique challenges to take full advantage of this architecture. Specifically, real time adaptation of capacity to efficiently support unicast and broadcast traffic, has not been addressed by any of these prior studies.

RoF in WLANs: Although some studies (e.g., [11, 43, 22, 12]) do talk about the feasibility and challenges of delivering WiFi services using RoF, they do not offer any concrete solution to our problem of catering to dynamic unicast and broadcast traffic demands.

SDN in WLANs: Studies such as [41, 50] propose a SDN-based WLAN architecture by decoupling control plane functions (power control, channel allocation, handoff etc.) from the data plane. However, since the decoupling is not

at the physical layer, these architectures lack some of the key benefits that come with our C-RAN-based architecture, e.g., the ability to add/remove capacity in real time and the support for network-wide broadcast.

Software-programmable C-RAN in cellular: The idea of employing re-configurable front-haul was first discussed in [31]. Subsequently in [46], the authors leveraged the re-configurable front-haul to conserve resource (BPU) usage in the processing cluster. We feel that this is the work most similar to ours in principle. However, the problem and the eventual solution discussed in [46] is specific to cellular and cannot be directly applied to WLANs due to vast technical differences between the two domains. Further, the particular optimization problem is different and thus is not applicable.

Reconfigurable networks in datacenter: Lastly, the ability to dynamically reconfigure a network, based on the input traffic demand has been explored in [32, 28]. However, these works do not consider 802.11 WLANs.

7. CONCLUSIONS

In this work, we propose AmorFi — a first-of-its-kind system that dynamically provisions WLAN capacity to handle spatiotemporal traffic changes. AmorFi’s architecture empowers WLANs with the ability to support network-level broadcast in its truest sense, thereby eliminating AP contention when delivering broadcast content. We evaluate AmorFi on a C-RAN testbed using detailed experimentation and large scale simulations. We show that AmorFi significantly outperforms traditional WLAN schemes owing to its software-defined access capability.

8. ACKNOWLEDGMENTS

We would like to thank our shepherd Samir Das and other anonymous reviewers for their insightful comments and feedback.

9. REFERENCES

- [1] 802.11aa-2012 - Amendment to IEEE Std 802.11(TM)-2012. <https://standards.ieee.org/findstds/standard/802.11aa-2012.html>.
- [2] N. Ahmed and S. Keshav. SMARTA: A Self-Managing Architecture for Thin Access Points.

- In *ACM CoNEXT*, 2006.
- [3] Cisco connected mobile experiences. http://www.cisco.com/c/en/us/td/docs/solutions/Enterprise/Borderless_Networks/Unified_Access/CMX.html.
 - [4] High density wireless networks. <http://community.arubanetworks.com/aruba/>.
 - [5] P. Bahl, M. Hajiaghayi, K. Jain, V. Mirrokni, L. Qiu, and A. Seberi. Cell Breathing in Wireless LANs: Algorithms and Evaluation. In *IEEE Transactions on Mobile Computing*, 2006.
 - [6] D. Brelaz. New methods to color the vertices of a graph. In *Communications of the ACM*, volume 24(2), 1979.
 - [7] I. Broustis, K. Papagiannaki, S. V. Krishnamurthy, and M. Faloutsos. MDG: Measurement-driven Guidelines for 802.11 WLAN Design. In *ACM MobiCom*, 2007.
 - [8] Benefits of Centralization in Wireless LANs. https://www.cisco.com/web/AP/wireless/pdf/Benefits_of_centralizedWLAN.pdf.
 - [9] Centralized Management and RF Security Moves Ariba to Aruba. http://www.arubanetworks.com/pdf/solutions/CS_ariba.pdf.
 - [10] R. Chandra, S. Karanth, T. Moscibroda, V. Navda, J. Padhye, R. Ramjee, and L. Ravindrananth. DirCast: A Practical and Efficient Wi-Fi Multicast System. In *IEEE ICNP*, 2008.
 - [11] G. Chang, C. Liu, and L. Zhang. Architecture and Applications of a Versatile SmallCell, Multi-Service Cloud Radio Access Network Using Radio-over-Fiber Technologies. In *IEEE ICC*, 2013.
 - [12] B. Charbonnier, H. Le Bras, P. Urvoas, Q.T. N’Guyen, M. Huchard, and A. Pizzinat. Upcoming perspectives and future challenges for ROF. In *IEEE report*, 2007.
 - [13] Yu-Chung Cheng, Mikhail Afanasyev, Patrick Verkaik, Peter Benko, Jennifer Chiang, Alex C. Snoeren, Stefan Savage, and Geoffrey M. Voelker. Automating Cross-Layer Diagnosis of Enterprise Wireless Networks. In *ACM Sigcomm*, 2007.
 - [14] Yu-Chung Cheng, John Bellardo, Peter Benko, Alex C. Snoeren, Geoffrey M. Voelker, and Stefan Savage. Jigsaw: Solving the Puzzle of Enterprise 802.11 Analysis. In *ACM Sigcomm*, 2006.
 - [15] Wireless lan design guide for high density client environments in higher education. http://www.cisco.com/c/dam/en_us/solutions/industries/docs/education/cisco_wlan_design_guide.pdf.
 - [16] Us convention centers. https://en.wikipedia.org/wiki/List_of_convention_centers_in_the_United_States.
 - [17] Time for Some New Acronyms for CRAN. <http://www.lightreading.com/mobile/fronthaul-c-ran/time-for-some-new-acronyms-for-cran/a/d-id/716159>.
 - [18] C-RAN and Future of Small Cells. <http://www.techmahindra.com/sites/blogs/C-RAN-and-Future-of-Small-Cells.aspx>.
 - [19] ENABLING CLOUD-RAN WITH MOBILE FRONTHAUL. <https://www.infinera.com/wp-content/uploads/2016/02/AN-Enabling-Cloud-RAN-with-Mobile-Fronthaul.pdf>.
 - [20] Benefits of Cloud RAN Virtualization Delayed, but Centralized RAN Saved Millions in 2014 OPEX. http://www.advfn.com/news_Benefits-of-Cloud-RAN-Virtualization-Delayed-but_65246303.html.
 - [21] Jeffrey Dean and Sanjay Ghemawat. MapReduce: Simplified Data Processing on Large Clusters. In *USENIX OSDI*, 2004.
 - [22] S. Deronne, V. Moeyaert, and S. Bette. WiFi Transmission in Radio-over-Fiber Systems: Performance of the IEEE 802.11n Aggregation Mechanism. In *ONDM*, 2013.
 - [23] Dicon Fiberoptics, Inc. http://www.diconfiberoptics.com/products/mems_multicast_optical_switches.php.
 - [24] Stadiums race to digitize:How sports teams are scrambling to keep Millennials coming to games. <http://www.techrepublic.com/article/how-sports-teams-are-scrambling-to-keep-\millennials-coming-to-games>.
 - [25] Delivering content with LTE Broadcast. http://www.ericsson.com/res/thecompany/docs/publications/ericsson_review/2013/er-lte-broadcast.pdf.
 - [26] Globecom2003 attendees. <https://www.cs.ucsb.edu/~almeroth/conf/stats/#globecom>.
 - [27] Varun Gupta, Craig Gutterman, Yigal Bejerano, and Gil Zussman. Experimental Evaluation of Large Scale WiFi Multicast Rate Control. In *CoRR*, 2016.
 - [28] Navid Hamedazimi, Zafar Qazi, Himanshu Gupta, Vyas Sekar, Samir R. Das, Jon P. Longtin, Himanshu Shah, and Ashish Tanwer. FireFly: A Reconfigurable Wireless Data Center Fabric Using Free-Space Optics. In *ACM SIGCOMM*, 2014.
 - [29] Intel-Processor. http://ark.intel.com/products/91750/Intel-Xeon-Processor-E5-2687W-v4-30M-Cache-3_00-GHz.
 - [30] Stadium App - Levi’s Stadium. <http://www.levisstadium.com/stadium-info/stadium-app/>.
 - [31] C. Liu, K. Sundaresan, M. Jiang, S. Rangarajan, and G. Chang. The Case for Re-configurable Backhaul in Cloud-RAN based Small Cell Networks. In *IEEE Infocom*, 2013.
 - [32] He Liu, Matthew K. Mukerjee, Conglong Li, Nicolas Feltman, George Papen, Stefan Savage, Srinivasan Seshan, Geoffrey M. Voelker, David G. Andersen, Michael Kaminsky, George Porter, and Alex C. Snoeren. Scheduling Techniques for Hybrid Circuit/Packet Networks. In *ACM CoNEXT*, 2015.
 - [33] Marriott marquis. <http://www.marriott.com/hotels/event-planning/business-meeting/sfodt-san-francisco-marriott-marquis/>.
 - [34] R. Murty, J. Padhye, R. Chandra, A. Wolman, and B. Zill. Designing High Performance Enterprise Wi-Fi

- Networks. In *Usenix NSDI*, 2008.
- [35] Low Latency Switching: The Race to Zero. http://www.netscout.com/uploads/2015/03/NetScout_WP_Low_Latency.pdf.
 - [36] Comparing costs of wireless LAN options. <http://www.computerworld.com/article/2574368/mobile-wireless/comparing-costs-of-wireless-lan-options.html>.
 - [37] Optical Zonu Corporation. <http://www.opticalzonu.com/standalone/oz816/>.
 - [38] LTE-U/LAA, MuLTEfire and Wi-Fi; making best use of unlicensed spectrum. <https://www.qualcomm.com/media/documents/files/whitepaper-making-the-best-use-of-unlicensed-spectrum.pdf>.
 - [39] Deploying very high density wi-fi: Design and configuration guide for stadiums. <http://c541678.r78.cf2.rackcdn.com/appnotes/bpg-highdensity.pdf>.
 - [40] Michael Sauer, Andrey Kobayakov, and Jacob George. Radio Over Fiber for Picocellular Network Architectures. In *Journal of Lightwave Technology*, 25, 3301-3320, 2007.
 - [41] J. Schulz-Zander, L. Suresh, N. Sarrar, A. Feldmann, T. Huehn, and R. Merz. Programmatic Orchestration of WiFi Networks. In *USENIX ATC*, 2014.
 - [42] S. Sen, N. K. Madabhushi, and S. Banerjee. Scalable WiFi Media Delivery through Adaptive Broadcasts. In *Usenix NSDI*, 2010.
 - [43] X. Shen, K. Xu, X. Sun, J. Wu, and J. Lin. Optimized Indoor Wireless Propagation Model in WiFi-RoF Network Architecture for RSS-Based Localization in the Internet of Things. In *IEEE MWP*, 2011.
 - [44] Vivek Srivastava, Shravan Rayanchu, Suman Banerjee, and Konstantina Papagiannaki. PIE in the Sky: Online Passive Interference Estimation for Enterprise WLANs. In *Usenix NSDI*, 2011.
 - [45] Cisco StadiumVision Mobile. <http://www.cisco.com/c/en/us/solutions/industries/sports-entertainment/stadiumvision-mobile.html>.
 - [46] K. Sundaresan, M. Y. Arslan, S. Singh, S. Rangarajan, and S. V. Krishnamurthy. FluidNet: A Flexible Cloud-based Radio Access Network for Small Cells. In *ACM Mobicom*, 2013.
 - [47] A. Vasan, R. Ramjee, and T. Woo. ECHOS - Enhanced Capacity 802.11 Hotspots. In *IEEE Infocom*, 2005.
 - [48] S. Vasudevan, K. Papagiannaki, C. Diot, J. Kurose, and D. Towsley. Facilitating Access Point Selection in IEEE 802.11 Wireless Networks. In *ACM IMC*, 2005.
 - [49] Enterprise Network Mobility: A TCO Analysis. http://www.arubanetworks.com/pdf/technology/whitepapers/wp_Lippis-TCO.pdf.
 - [50] Y. Yiakoumis, M. Bansal, A. Covington, J. van Reijendam, S. Katti, and N. McKeown. BeHop: a testbed for dense WiFi networks. In *ACM WinTECH*, 2014.