

Scalable WiFi Multicast Services for Very Large Groups

Yigal Bejerano*, Jaime Ferragut[‡], Katherine Guo*,
Varun Gupta[†], Craig Gutterman[†], Thyaga Nandagopal⁺, Gil Zussman[†]

[†] Electrical Engineering, Columbia University, New York, NY, USA.

* Bell Labs, Alcatel-Lucent, Murray Hill, NJ, USA.

[‡] Mobile Networks Department - CTTC, Barcelona, Spain.

+ NSF, Arlington, VA, USA.

Abstract—IEEE 802.11-based wireless local area networks, referred to as WiFi, have been globally deployed and the vast majority of mobile devices are currently WiFi-enabled. While WiFi has been proposed for multimedia content distribution, its lack of adequate support for multicast services hinders its ability to provide multimedia content distribution to a large number of devices. We propose *AMuSe*, a scalable and adaptive interference mitigation solution for WiFi multicast services which is based on accurate receiver feedback and that incurs a small control overhead. Specifically, we develop a scheme for dynamic selection of a subset of the multicast receivers as feedback nodes, which periodically send information, such as channel quality or received packet statistics, to the multicast sender. This feedback information is used by the multicast sender to optimize the multicast service quality, e.g., by dynamically adjusting the transmission bit-rate. Our proposed solution does not require any changes to the standards or any modifications to the WiFi devices. We have implemented the proposed solution in the *ORBIT* testbed and evaluated its performance in large groups with approximately 250 receivers, both with and without interference sources. Our on-line experiments demonstrate that our system provides practical multicast services that can accommodate hundreds of receivers.

Index Terms—802.11, WiFi, Multicast, Feedback Mechanism.

I. INTRODUCTION

Recent years have witnessed a rapid growth of mobile devices such as smart-phones and tablets equipped with wireless local area network (WLAN) interfaces that use WiFi standards [1]. While these devices allow users to access the Internet anywhere anytime, it is not straightforward to serve rich multimedia content, such as video streams, when users are clustered in crowded areas, due to a combination of high bandwidth requirements and shortage of wireless spectrum. The inability to serve this growing demand for multimedia content using limited resources in crowded areas has prompted several solutions by both industry and academia. Many of these solutions [2]–[4] are typically based on dense deployment of access points (APs) for providing dedicated content delivery to each user. Such solutions, besides requiring considerable capital and operational expenditure, may not meet user expectations, due to extensive interference between

adjacent cells.

In this paper, we present a light-weight solution for scalable delivery of rich multimedia content to a large number of users in a small geographical region by means of WiFi multicast. Our solution is an attractive method for delivering live video content to a large user population that share common interests. For instance, in a sports arena, our approach can be used for providing simultaneous video feeds of multiple camera angles.

Current state of the art solutions use IEEE 802.11 leveraging either unicast and multicast data delivery. Commercial solutions [3], [4] rely on streaming content to individual users. With standards such as 802.11ac promising speeds up to 800 Mbps per user using multi-user MIMO, it is theoretically possible to serve video streams to hundreds of users. However, recent studies [5], [6] throw cold water on this promise. Large number of neighboring APs lead to hidden terminal problems and this, coupled with increased interference sensitivity stemming from channel bonding, makes the entire solution interference limited. Extrapolating from studies on 802.11n [5], [6], it seems that 802.11ac-based unicast to multiple clients may not be able to support more than a hundred users, assuming all of them have 802.11ac capable devices.

Standard WiFi broadcast/multicast frames are transmitted at a fixed and low bit-rate without any feedback. This raises several known reliability and efficiency issues. High packet loss due to interference and the hidden node problem can significantly degrade service quality, while transmission at low bit-rates leads to low network utilization. There are numerous studies in the literature that propose solutions for overcoming these limitations from two aspects. One is aimed to reduce overhead of feedback information to the multicast sender. The other is to improve message reliability based on available feedback information. However, WiFi multicast services are rarely used by practical content delivery applications due to lack of scalability, reliability, or standard compliance.

A. Our Contribution

We present the *AMuSe* (Adaptive Multicast Services) system for providing scalable and efficient delivery of mul-

timedia content with low communication overhead to a large number of WiFi receivers, also called nodes. In particular, AMuSe enables the APs to transmit multicast traffic at the highest possible bit-rate, while ensuring high *packet delivery ratio* (PDR) for the vast majority (over 95%) of the nodes¹.

AMuSe is based on the observation that a cluster of adjacent nodes experience similar channel quality and interference patterns [8]. AMuSe dynamically divides the nodes into a few clusters based on distance between nodes. In each cluster, one node is selected as the *feedback (FB) node* and it reports to the AP about its own service quality, e.g., channel quality and lost packet statistics. The AP, in response, may decide to report back to the source, adjust the forward error correction (FEC), adapt the transmission bit-rate, retransmit lost packets, or execute a combination of the above.

AMuSe supports a simple and efficient *FB node selection process* that requires only a few nodes to send feedback messages, which results in low communication overhead. The process strikes a balance between the number of FB nodes, the accuracy of the feedback and the system convergence time by controlling the maximal radius of the clusters, denoted as D . The scheme ensures that every node is at most within distance D away from a FB node with similar or weaker channel quality. To ensure sparse FB node density, any pair of FB nodes are at least D distance apart. This is a variant of the well known *minimal independent dominating set* problem [9]. Although this problem is NP-hard, our distributed algorithm can find a solution with small constant approximation ratio, when paired with a widely-used signal propagation model. AMuSe offers the following salient features:

- 1) WiFi compatible — The scheme can be deployed with any standard WiFi device, as application layer protocol deployed on top of the WiFi interface of mobile devices.
- 2) Efficient wireless resource utilization — AMuSe efficiently selects a small number of FB nodes that produce only limited communication overhead while accurately reflecting the packet delivery ratios among all nodes.
- 3) Scalable solution — The partition into clusters depends only on the cluster radius and not the node density, therefore, AMuSe scales as a function of the geographical area.

We use the large-scale ORBIT testbed [10] for the evaluation of AMuSe, where each experiment involves approximately 250 WiFi nodes. One of our observations is that during any experiment, some nodes, called here as *abnormal nodes*, suffer from low PDR, even when the AP is using low transmission bit-rate and there is no external interference. Furthermore, this set of abnormal nodes varies across experiments. These nodes pose a considerable challenge, since they cause low network utilization, although nearly all nodes can benefit from much higher transmission bit-rate. In order to address this, we equip AMuSe with an abnormal node detection mechanism that separates abnormal nodes from normal FB nodes. Paired

¹We assume that the flow contains some redundancy (e.g. FEC) across packets [7] for dealing with limited amount of losses.

TABLE I
MULTICAST: FEATURES OF RELATED WORK

	Scalable	Reliable	Standards Compatible	Low cost
Basic 802.11 multicast	✓	x	✓	✓
Unicast/ Individual Feedback	x	✓	✓	✓
LBP-ACK	✓	x	x	✓
Pseudo Broadcast	✓	x	✓	✓
LBP-NACK	✓	x	x	✓
Cellular multicast	✓	x	✓	x
AMuSe	✓	✓	✓	✓

with a simple rate adaptation algorithm at the AP, we demonstrate that AMuSe is able to maximize transmission bit-rate while ensuring PDR of 90% for more than 95% of the nodes.

In the rest of the paper, we describe our network model and our objectives in Sections III and IV respectively. We present the design of AMuSe in Section V, our offline and online experimental results in Sections V-C and VII respectively before concluding in Section VIII.

II. RELATED WORK

Nearly all of the solutions to improve multicast services are based on integrating automatic repeat request (ARQ) mechanisms into the protocol architecture [11]–[15], adding forward error correction (FEC) packets to the multicast stream [16], [17], or both [18]. Other studies propose rate adaptation mechanisms for improving network utilization [19].

In all cases, a key requirement is having appropriate feedback from receivers regarding their service quality. These feedback gathering mechanisms can be classified into four main categories: (1) *Individual Feedback* from multicast receivers, (2) Leader-Based Protocol with acknowledgements (LBP-ACK), (3) *Pseudo-Broadcast*, and (4) *Leader-Based Protocol* with negative acknowledgements (LBP-NACK). Table I summarizes the features of these approaches.

Individual Feedback mechanisms require all receivers to send acknowledgements of received packets. With *More Reliable Groupcast* (MRG) [1], [20] from IEEE 802.11 working group, each receiver transmits a bit-map of correctly received packets. Using this feedback, the sender determines lost packets and retransmits them to the group. This approach offers reliability but incurs high feedback overhead with large groups. The other three approaches reduce this overhead as follows.

The LBP-ACK approach [14]–[16], [18], [20], [21] provides scalability by selecting a subset of the receivers to provide feedback. The Pseudo-Broadcast approach [11], [12], [21], [22], converts the multicast feed to a unicast flow and sends it to one leader, typically, the receiver with the weakest channel. The leader acknowledges the reception of the unicast flow. The other receivers receive packets by listening to the channel in promiscuous mode. The LBP-NACK approach [13], [19], [23] improves Pseudo-Broadcast by allowing the other receivers to send NACKs for lost packets. After receiving the ACK from the leader, the sender can infer successful transmission to all receivers since an NACK would collide with the leader’s ACK.

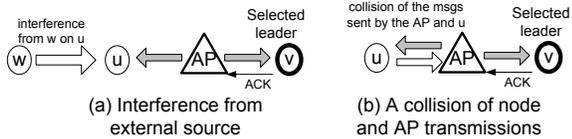


Fig. 1. Unreliable packet delivery by the LBP and the Pseudo-Broadcast approach.

With LBP-ACK and Pseudo-Broadcast, the selection of the leader(s) or subset of the receivers to provide feedback, can compromise service reliability. In Fig. 1-(a), the leader v acknowledges a packet on behalf of node u , even though node u suffers from external interference that prevents correct reception of the packet. In Fig. 1-(b), the node u might have an uplink transmission collide with the multicast packet from the AP, but since the leader correctly receives the multicast packet, the AP thinks the transmission has succeeded.

The LBP-NACK approach not only requires changes to the standard, but also suffers from lack of reliability. As a non-leader cannot reply with a NACK if it cannot identify a corrupted packet. Furthermore, due to capture effect the AP may be able to decode the ACK and ignore NAK messages.

Orthogonal to WiFi, cellular network providers have recently considered femto-cell based deployments in dense urban areas to provide additional capacity, and LTE-based multicast solutions have been proposed [24] for the scenarios considered in this paper. While these provide another mode of content delivery, the basic issues with respect to obtaining receiver feedback exist in such systems, in addition to being expensive to deploy due to the cost of licensed spectrum.

Current WiFi multicast mechanisms requiring receiver feedback do not consider abnormal nodes. Therefore network utilization suffers with the presence of abnormal nodes. The basic 802.11 multicast mechanism without any node feedback simply sets the transmission bit-rate to the lowest rate. Cellular networks also operate without any node feedback and set the transmission bit-rate to be a low value ignoring the presence or absence of nodes near the edge of the cell. Any multicast mechanism without node feedback results in perpetual low network utilization.

In summary, one or more of the following weaknesses prevent existing schemes from being successful: (a) requirement for feedback from large numbers of receivers, (b) ignorance of the source of interference-related packet loss, (c) lack of feedback information leads to low network utilization, (d) requirement for changes to standard WiFi protocol, or (e) high cost of deployment. This motivates our desire for a scalable solution that improves reliability of multimedia content delivery based on WiFi deployments.

III. NETWORK MODEL

We consider an IEEE 802.11 WLAN, termed WiFi Network, with multiple *access points* (APs) that serve a very dense mobile population. Due to shortage of orthogonal wireless channels, adjacent APs may use the same channels. Conse-

quently, a transmission in one cell may interfere with transmissions in adjacent cells. We consider transmissions from both mobile devices and APs outside of the cell as potential sources of interference. We focus on a single cell with a given AP, The AP sends a multicast flow to a large *multicast group* of mobile devices in its transmission range, referred to as *nodes* or *receivers*.

We follow the model where a node may report to an AP or a multicast server about its experienced service quality, e.g., channel quality and lost packet statistics. The AP or the multicast server, in response, may decide to adjust the FEC, adapt the transmission bit-rate, retransmit lost packets, or execute a combination of the above. In practice, the AP and the multicast server are two separate logical entities. These functions may reside in the multicast server or the AP and in multiple network layers. However, only the AP is responsible for adjusting the transmission bit-rate. Therefore we use AP to represent the combination of AP and multicast server for the rest of the paper.

At any given time each node is associated with a single AP and nodes are assumed to have a quasi-static mobility pattern. In other words, nodes are free to move from place to place, but they tend to stay in the same physical locations for several minutes or more. This is a reasonable assumption for various crowded venues, such as sports arenas or transportation hubs. We assume that mobile devices can estimate their locations, e.g., by using one of the methods in [25] with an accuracy of a few meters, and also determine if they are *static*² or *moving*.

IV. OBJECTIVES

Our challenge is to design a scalable WiFi-based multimedia content distribution system for a very large number of nodes. Our objective can be formulated as follows:

Objective: *Designing an efficient WiFi multicast mechanism with minimal overhead that transmits the multicast traffic at highest possible bit-rate, while ensuring high packet delivery ratio (PDR) for majority of the nodes.* More specifically, ensuring that at least X percentage of the nodes, e.g., $X = 95\%$, experience PDR above a given threshold³ denoted by H , e.g., $H = 90\%$.

We rely on the following observation reported in related studies [8]:

Observation: *A cluster of adjacent nodes experiences similar channel quality and suffers from similar interferences. Hence, a node v with a lower channel condition than its adjacent neighbors can represent the service quality observed by the nodes in the cluster. Furthermore, if a packet is correctly received by v then, with high probability, it is correctly received by nodes within the cluster as well.*

Based on this observation, we divide the nodes into clusters of adjacent nodes and select a single *feedback (FB) node* in each cluster to represent all nodes in the cluster, e.g., in order to report on the PDR of the multicast traffic. These

²We consider a node static if its movement is restricted to a few meters.

³We assume that the flow contains some redundancy (e.g. FEC) across packets [7] for dealing with limited amount of losses.

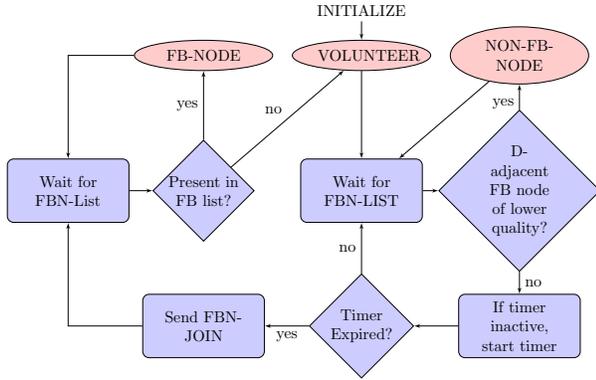


Fig. 2. State diagram of AMuSe FB node selection algorithm at each node. All nodes initialize in the VOLUNTEER state.

reports can be used by the AP for improving the quality of the multicast service by using the following measures proposed in the literature: retransmitting lost packets, adding FEC, or adapting the transmission bit-rate.

Rather than considering the appropriate actions for improving the multicast service quality at the AP, our focus is the design of an efficient feedback mechanism and the FB node selection process. Our feedback mechanism finds a small set of well-distributed FB nodes that provide reports of the service quality experienced by all receivers to a desired accuracy.

V. THE AMUSE SYSTEM

This section provides a general overview of the AMuSe system. We define two nodes to be *D-adjacent* if they are separated by distance at most D for any given D . In order to find a small set of FB nodes that can provide accurate reports, all nodes in our system satisfy the following requirements.

Requirements:

- Each node should be D -adjacent to a FB node.
- An FB node must have similar or weaker channel quality than its D -adjacent nodes.
- Any two FB-nodes cannot be D -adjacent to each other.

We consider several channel quality metrics, including *received signal strength* (RSS), *signal-to-noise ratio* (SNR) and *packet delivery ratio* (PDR), and these are compared experimentally in Section V-C.

A. The Feedback Node Selection Algorithm

We propose a semi-distributed process for selecting the FB nodes (FBNs), where some nodes volunteer to serve as FB nodes, and the AP selects the best candidates. If node location information as well as their observed channel quality is known, then the AP can easily select the ideal set of FB nodes. However, this is not feasible in practice for large groups. Therefore, we seek to minimize the number of nodes that send their information to the AP as part of the FB node selection process, while ensuring a small set of FB nodes that meet the above requirements.

The AP periodically (e.g., once every $\tau_{AP} = 500$ ms in our experiments) sends an FBN-LIST message with a list of FB nodes. Each entry in the FBN-LIST contains the node id⁴, its reported location⁵, its reported channel quality, and a measure of the packet delivery ratio⁶.

In AMuSe, each node is in one of three states:

FB-NODE State - A static node that has been selected as FB node.

VOLUNTEER State - A static node that is not aware of any D -adjacent FB node with lower or similar channel quality and can serve as a FB node.

NON-FB-NODE State - A regular node that either is in a moving state or is aware of a D -adjacent FB node with similar or lower channel quality.

When a node v joins the system, it is in the VOLUNTEER state. The node waits for an FBN-LIST message, and checks if there are any D -adjacent FB nodes in this list with similar or weaker channel quality. If there are any such nodes, node v switches to the NON-FB-NODE state and records the list of D -adjacent FB nodes in the FBN-LIST message with similar or weaker channel quality. If there are no such nodes, node v starts a random back-off timer for a period chosen in the interval $[0, T]$ (our experiments use the maximum receiver back-off timer $T = 5$ seconds). During this countdown, if node v learns of a D -adjacent FB node from a FBN-LIST message, then it cancels its countdown, and switches to a NON-FB-NODE state. Otherwise, upon expiry of the timer, it will send a FBN-JOIN message to the AP, and waits to see if its ID appears on the next FBN-LIST. The FBN-JOIN message contains the node ID, node location and observed channel quality. If node v appears on the FBN-LIST, it switches to the FB-NODE state. If not, it repeats the back-off process again until it leaves the VOLUNTEER state. At any time, upon receipt of a FBN-LIST message, if an FB-node v does not find itself on the FBN-LIST, it has to leave the FB-NODE state. If it finds a D -adjacent FB node in the list, then it switches to the NON-FB-NODE state, otherwise it moves to the VOLUNTEER state.

An important property of this volunteering process is that the FB node selection is done in a semi-distributed manner, since a node volunteers to serve as FB node only if there is no other FB-node in its vicinity with weaker channel quality. Thus, the responsibility of the AP is only to resolve conflicts when several D -adjacent nodes volunteer simultaneously and to prune unnecessary FB nodes. Consequently, after receiving FBN-JOIN messages and just before sending a FBN-LIST message, the AP runs node pruning algorithm, described later in this section, to decide on which nodes to serve as FB nodes.

Each FB-node periodically (e.g., once every $\tau_{FB} = 500$ ms in our experiments) sends REPORT messages to update the

⁴Nodes can be assigned temporary virtual IDs to maintain privacy.

⁵We are aware that relying on a user to be truthful about their location or channel quality could lead to denial-of-service attacks, but these can be mitigated using other means and we shelve this orthogonal discussion for now.

⁶This can be easily changed to report the last acknowledged packet sequence number to support finer granularity of message reliability.

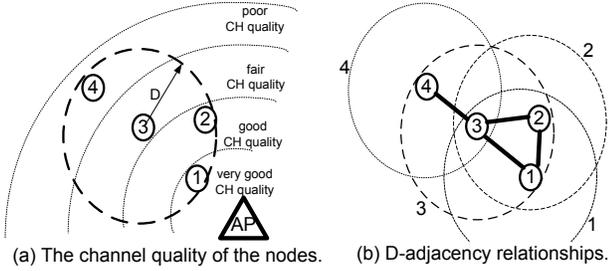


Fig. 3. Example WLAN with 4 receivers.

AP about the channel and service quality that the node, and thus its representative cluster experiences. If the AP does not receive any message from one of the FB nodes for a given duration, for example, $3\tau_{FB}$ (used in our experiments), then the AP removes it from the list of FB nodes. We present the state transition diagram for each node in Fig. 2.

When a new FB node is selected for a previously known cluster, the AP sets the last acknowledge packet for that FB node to be the same as that of the previous FB node for that cluster. If the FB node is selected for a new cluster (i.e., a new geographical region), the last acknowledged packet is set to the last multicast stream packet sent by the AP prior to the FBN-LIST message, thus allowing the new FB-node to begin acknowledging the stream from this point onwards.

A few aspects of the AMuSe system are worth pointing out here. First, we do not require the nodes to listen to all the traffic on the network. All they have to do is listen to the AP, on the multicast group address. This conserves energy at the receivers. Second, we do not require the location information for nodes to be very precise. As mentioned in Section III, coarse granularity is acceptable, as long as the accuracy is in the order of few meters, which has been demonstrated by some studies as feasible and practical. Third, we provide for variable levels of reliability by fine-tuning the combination of AP reporting frequency τ_{AP} , the receiver reporting frequency τ_{FB} , the maximum receiver back-off timer T , and the node adjacency distance D , with more reliable and frequent reports leading to more overhead. Instead of a single control, we provide for multiple control knobs, giving greater flexibility to the operator to decide on the type of service support based on the nature of multicast streams.

B. Illustrative Example

Consider the WLAN shown in Fig. 3-(a) with a single AP and four receivers. Assume that numbers labeling the nodes denote the IDs and the order in which they join the multicast service at this AP. There are four different channel quality levels: very good, good, fair and poor as experienced by node 1, 2, 3, and 4 respectively. Fig. 3-(b) shows a circle with radius D around every node, say node v , where each node, u , inside the circle of v is D -adjacent to node v . Hence, nodes u and v are considered neighbors to one another.

In this example, we demonstrate the importance of all three requirements mentioned at the beginning of this section,

especially (c), on the quality and density of the set of FB nodes. Assume first that the FB nodes have to meet only requirement (a) and (b), but not (c). Under these guidelines, at the moment each node joins the multicast, it has a weaker channel quality than all its neighbors and therefore it is selected as a FB node. At the end the system contains four FB nodes. It is easy to see that this approach does not scale for large groups.

Now, let us assume that requirement (c) is enforced. Right after a node joins the system, the set of FB nodes is optimized. When node 1 joins, it becomes the FB node. After node 2 joins, node 2 becomes the FB node, while node 1 becomes a non-FB node because of (c). After node 3 joins, it becomes a FB node while both node 1 and 2 become non-FB nodes because all three nodes are D -adjacent to one another. After node 4 joins, it becomes a FB node, while node 3 becomes a non-FB node. In addition, node 2 becomes a FB node again. Notice that node 2 switches state twice, after node 3 and 4 joins respectively. However, after each node joins the multicast, the set of FB nodes is optimal.

This example shows that while our FB node selection algorithm satisfying all three requirements it may cause churn as nodes join and leave the FB node set, the selected set of FB nodes is near-optimal when the set of nodes receiving the multicast do not change, as we show next.

C. The Node Pruning Algorithm

As described above the FB node selection process ensures that every receiver is D -adjacent to a *candidate* node with similar or weaker channel condition, where the candidate list contains the current FB nodes as well as the nodes in the VOLUNTEER state. Thus, it is the responsibility of the AP to trim unnecessary candidates for having a small set of FB nodes such that any pair are not D -adjacent.

The problem of finding the minimal set of FB nodes that meets the three requirements above is a variant of the *minimal dominating set problem*, which is a known NP-complete problem also in the case of unit disk graph [9]. Below we present a heuristic algorithm that selects near optimal set of candidates that meet our three requirements.

The heuristic algorithm: The AP creates a list L of the candidates sorted in increasing order according to their channel quality. Then, it iteratively selects the first candidate v in L as a FB node and remove v and all its D -adjacent nodes from L . The algorithm ends when L is empty.

Let F denotes the FB nodes selected by the heuristic algorithm and OPT denotes the optimal (minimal) set of FB nodes among all nodes, our algorithm ensures the following property:

Theorem 1: $|F| \leq 5 \cdot |OPT|$. If the channel quality is monotonic decreasing function with the distance from the AP then $|F| \leq 3 \cdot |OPT|$

The proof is based on Lemmas 3.1 and 3.3 in [9] and it is omitted due to space limitation.

Stability vs. optimality trade-off: As illustrated in Section V-B, a naive implementation of the heuristic algorithm

TABLE II
EVALUATION PARAMETERS

Parameter	Definition
LQ_i	Link Quality of node i with the AP.
P_i^{vec}	A vector of the packets received by node i .
(x_i, y_i)	(row, column) location of node i .
TX_{AP}	Broadcast/Multicast transmission rate at the AP.

may cause churn of FB nodes, which obstructs system stability. Since node pruning is done by the AP, the algorithm can be easily modified to prevent churn, for instance by giving higher priorities to already selected FB nodes or relaxing the distance constraint between FB nodes. Although, striking a proper balance between system stability and optimality of the FB node selection is a central topic in our system design, it is beyond the scope of this paper.

VI. OFF-LINE EXPERIMENTS

We have validated our scheme using the 400-node ORBIT testbed [10], and describe these experiments in this section.

We use the *link quality* (LQ) metric⁷ reported by a node's WiFi card as representative of its observed *received signal strength* (RSS). We first consider a set of hypotheses used to validate the observation in Section IV and thereby, our system design.

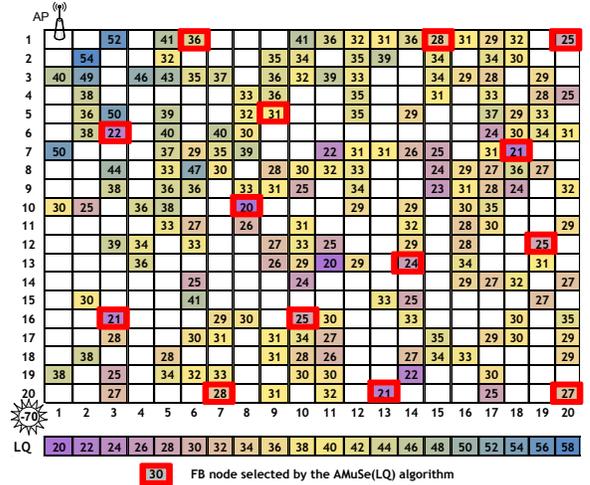
- H1:** There is some correlation between *packet delivery ratio* (PDR) and *link quality* (LQ) observed by a node.
- H2:** Clustered nodes experience similar LQ and similar PDR.
- H3:** Clustered nodes suffer from similar interference.
- H4:** A node with lower LQ than its neighbors may serve as a good FB node for representing the PDR observed by its neighbors.

In addition, we also investigate the parameter choices for cluster radius (represented by the D -adjacency parameter, D) and the sensitivity of our scheme to errors in reported node locations.

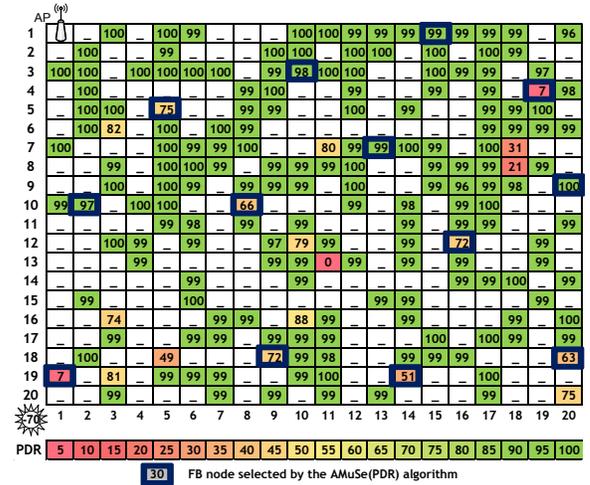
A. The ORBIT Testbed and Experiment Settings

The ORBIT testbed [10] is a dynamically configurable grid of 20×20 (400 overall) nodes with 802.11 radio. The grid separation between nodes is 1 meter and in addition, the testbed provides a noise generator with 4 noise antennas at the corners of the grid whose attenuation can be independently controlled, permitting the emulation of a richer topology. In order to avoid performance artifacts stemming from a mismatch of WiFi hardware and software, we select the subset of nodes equipped with *Atheros 5212/5213* wireless cards with *ath5k* wireless driver. Furthermore, we remove unresponsive nodes in the grid before every experiment which results in approximately 250 nodes for each experiment.

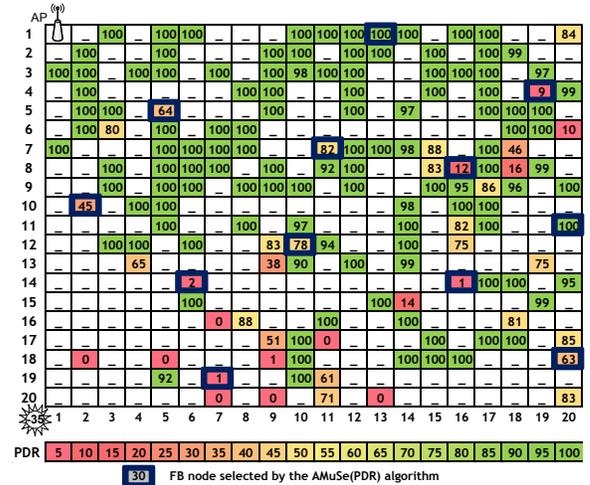
⁷Although LQ is not a standard measurement metric, we observed that the reported LQ by the Atheros chipsets indicated the RSS in db normalized to a reference value of -110dBm (the thermal noise).



(a) Link Quality Heatmap, noise = -70 dBm.



(b) PDR Heatmap, noise = -70 dBm.



(c) PDR Heatmap, noise = -35 dBm.

Fig. 4. Link Quality (LQ) and PDR heatmaps at the AP with transmission bit-rate of 12 Mbps and noise level of -70 dBm and -35 dBm. The FB nodes are highlighted with a thick border in red in the LQ heatmap and in blue in the PDR heatmap for $D = 6$ meters. Empty locations represent nodes that did not produce LQ or PDR reports and they are excluded from our experiments. Nodes with $PDR = 0$ are active nodes that reported LQ values but were unable to decode packets. These nodes are excluded from the FB node selection process.

We have implemented the AMuSe system as an application layer program (for clients and the AP), running on all nodes. The node at the corner (1, 1) serves as a single multicast AP, configured in master mode, and it uses channel 40 of 802.11a⁸ to send a multicast UDP flow with a transmission power of 1 mW= 0 dBm. The other nodes are the multicast receivers, configured in managed mode. This means that in practice our experiments consider *less than a quarter of the transmission range of an AP*. Each UDP packet is 1400 bytes in payload length and contains sequence numbers to detect packet loss.

Every node keeps track of the parameters described in Table II, which we process off-line after each experiment. The *packet delivery ratio* (PDR) value of each node i is calculated from its P_i^{vec} vector.

B. Experiment Description

We now describe the three types of experiments conducted. The first two types of experiments are mainly used to validate our hypotheses presented earlier in this section. The third type is used to demonstrate how a rate adaptation algorithm works with the feedback selection mechanism. Only a representative subset of all results are reported here due to space limitation. Each node is identified by its (row, column) location.

Different Bit-rates: We fix the AP multicast transmission bit-rate, denoted by TX_{AP} , to different values allowed by the card (6, 9, 12, 18, 24, 36, 48, 54 Mbps), each bit-rate for a duration of 10 seconds. We repeat these experiments 10 times at different times of the day without any external noise.

Different Noise Levels: We fix the AP multicast transmission bit-rate to 12 Mbps and turn on the noise generator near node (20, 1). The noise generator is configured to provide AWGN noise for the entire spectrum of channel 40. Starting with -80 dBm (very low noise), we vary noise power in steps of 5 dBm up to -5 dBm (very high noise).

With Rate Adaptation: We run experiments with FB node selection and a simple rate adaptation algorithm turned on at the AP to demonstrate the practicality of our design and evaluate system performance.

Fig. 4 presents three sample heatmaps of one run of the experiments, when $TX_{AP} = 12$ Mbps and external noise of -70 dBm and -35 dBm generated near node (20,1). Each heatmap shows the active nodes used in the experiment and either the LQ or PDR values that they experienced, in addition to the FB nodes that the AP has selected with D -adjacency parameter of 6 meters. Nodes marked with thick red or blue border are FB nodes selected by the AMuSe scheme. Nodes with $PDR = 0$ are active nodes that reported LQ values but unable to decode packets in the experiment run. For example, in Fig. 4(a) and 4(b) for the run with noise level at -70 dBm, node (13,11) with $PDR = 0$ and $LQ = 20$ is such a node. These nodes are excluded from the FB node selection algorithm.

An interesting observation is that a selected FB node v may have higher PDR (or LQ) than an adjacent non-FB node, say

u . Such a situation results from the independent-set property of the selected FB nodes and it may occur if u is D -adjacent to another FB node with even lower PDR (or LQ). For instance, in Fig. 4(b) Node (7, 13) with PDR of 99% was selected as FB node although it has a neighbor, Node (7, 11), with PDR of 80%. The reason is that Node (7, 11) is 6-adjacent to FB node (10, 8) with PDR of 66%.

C. Hypotheses Evaluation

We turn to evaluate our hypotheses based on the information collected from the first two types of experiments in Section VI-B.

H1 - Correlation between PDR and LQ: Figs. 5(a)-5(d) demonstrate the correlation between the PDR of a node with respect to its LQ for different transmission bit-rates without external noise, while, Fig. 5(e) demonstrates the correlation between the PDR of a node with respect to its distance from the AP with a transmission bit-rate of 48 Mbps. PDR values are close to 100% for almost all nodes for bit-rates up to 24 Mbps (Fig. 5(a)-5(b)). Some degradation of the PDR values is observed for bit-rates of 36 Mbps (Fig. 5(c)) and even higher variance of PDR values are seen for 48 Mbps (Fig. 5(d)) and above.

Fig. 5(d) shows that the correlation between the PDR and LQ is not very strong, suggesting that nodes with the same LQ value may have significantly different PDR. Similarly, Fig. 5(e) illustrates weak correlation between the PDR of a node and its proximity to the AP (with $TX_{AP} = 48$ Mbps), and some of the nodes adjacent to the AP suffer from low PDR. For instance, Fig. 5(e) shows that one of the nodes with distance of 5 meters from the AP suffers from PDR of 25%. This observed variation of PDR with LQ as well as variation of PDR with distance to the AP is consistent with prior work, e.g., [26], [27] and [28].

H2 - Clustered nodes experience similar LQ and PDR: We calculate the standard deviation (STD) of LQ and PDR in each cluster of radius of 1.5 meters on the grid, where each cluster contains a FB node and all its neighbors (between 1 to 8 nodes). Histograms of the distribution of the LQ and PDR STD in different clusters are shown in Fig. 5(f) and Fig. 5(g) respectively. We measure the same distributions in the presence of various noise levels, and plot the results in Fig. 5(h) and Fig. 5(i), respectively. We expect the STD across clusters to be a good measurement of how similar the PDR and the LQ values are for AMuSe .

Comparing Fig. 5(f) and Fig. 5(h), we see that the LQ STD is very similar across all the bit-rates regardless of the noise levels, as expected. Nearly 75% of the clusters have LQ STD between 0–4 db, which confirms our assumption that adjacent nodes experience similar LQ (and similar RSS), and that LQ metrics do not log the level of interference seen in the system.

We now consider the distribution of the PDR STD values. Fig. 5(g) shows that with $TX_{AP} \leq 36$ Mbps, only very few clusters show significant deviations ($> 5\%$) in PDR, as most nodes have PDR above 99%. We see significant variability of the PDR at higher bit-rates. The addition of interference,

⁸We observed that channel 40 at the 5 GHz band suffers from less external interferences on the ORBIT grid than the channels at 2.4GHz band.

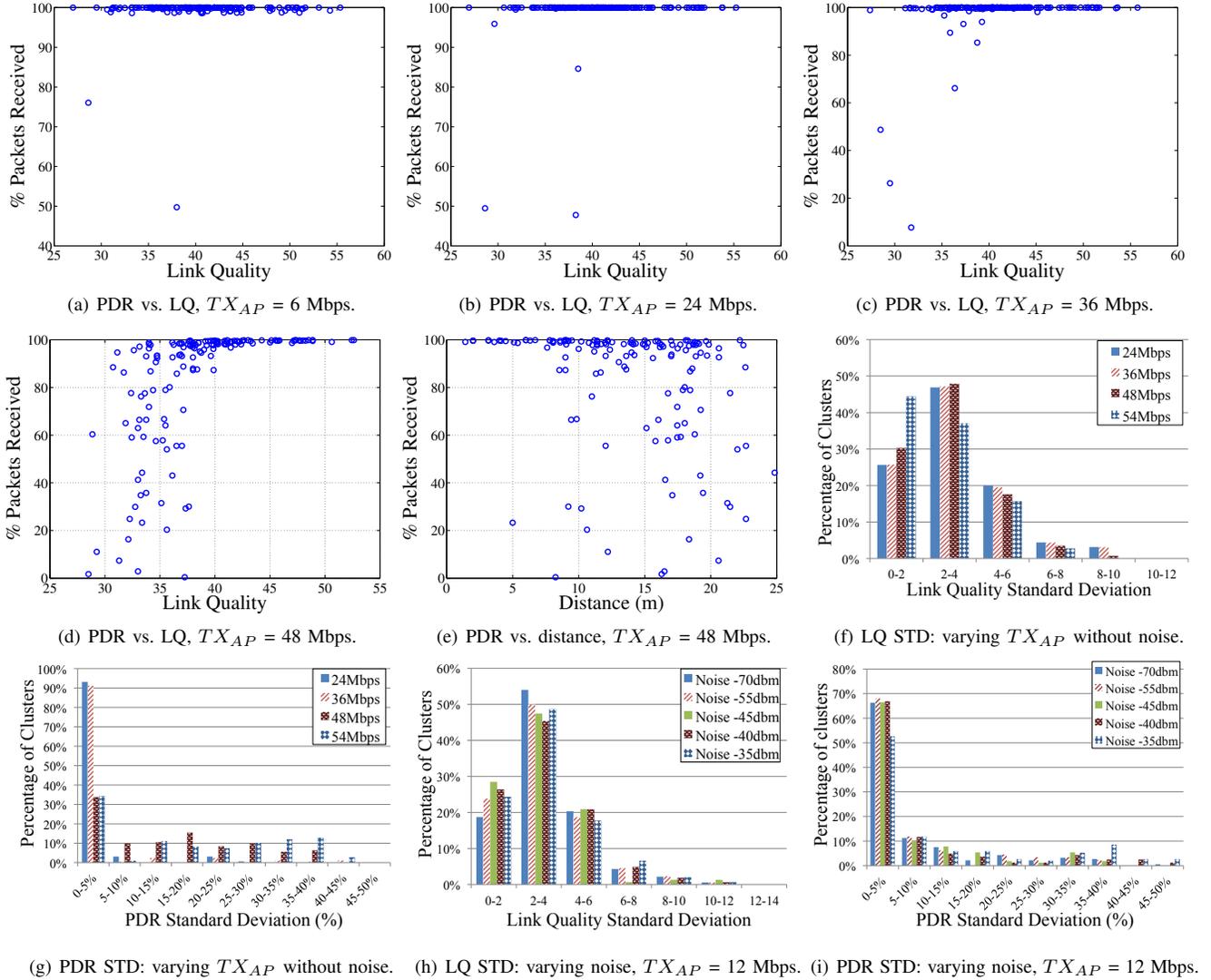


Fig. 5. Experimental results for evaluating hypotheses H1-H3 and abnormal nodes.

shown in Fig. 5(i), introduces noticeable deviations ($> 5\%$) in PDR across nearly one-third of the clusters.

To understand this, we revisit the heatmaps in Fig. 4(c). It is clear that the PDR values are decreasing for nodes near the bottom-left corner where the noise generator is located. The nodes which report a zero PDR value are the ones that are unable to either decode the AP beacons or receive the multicast flow. The former nodes are not shown in the heatmap (and are not included in the variance calculations), while the latter ones are shown in red with a zero value. This becomes very noticeable with higher noise levels. This explains the high levels of PDR variance observed in Fig. 5(i).

H3: Clustered nodes suffer from similar inference: Fig. 4 and Fig. 5(i) demonstrate that external noise has only local effect, which validates the need for a well-distributed set of FB nodes to report on interference experienced by receivers.

Furthermore, our experiments demonstrate that increasing TX_{AP} has an impact on all receivers, and that beyond a

certain bit-rate, the PDR of many nodes drops below 90%, as shown in Fig. 5(d) and Fig. 5(e). Thus, it is critical to set the TX_{AP} to the appropriate value in order to improve the efficiency of the multicast service.

D. Abnormal Nodes

In general, we call a node with a low PDR as abnormal. In our experiments, we define a node to be *abnormal* if its PDR is below the abnormal threshold $H = 90\%$. In this section, we study the number of abnormal nodes as a function of the TX_{AP} and the link quality (LQ). Fig. 5(a) - 5(d) show how PDR varies with LQ for each node in a single experiment run with TX_{AP} bit-rates of 6, 24, 36 and 48 Mbps, respectively. Results from all values of TX_{AP} (including ones not shown here) show that the number of abnormal nodes increases with the increase of TX_{AP} .

In Fig. 5(a) - 5(c), PDR values are close to 100% for the vast majority of the nodes for bit-rates up to 36 Mbps.

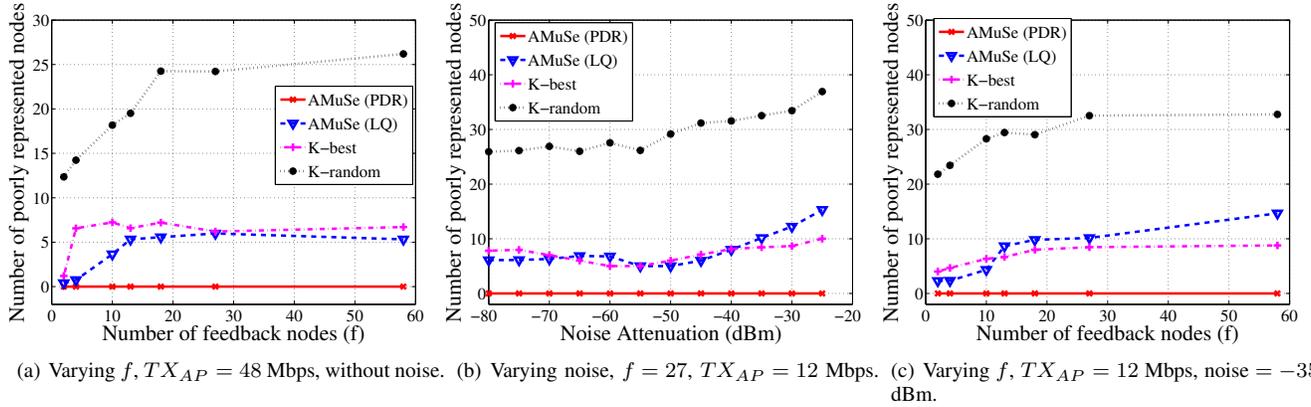


Fig. 6. The average number of poorly represented nodes under various scenarios.

However, Fig. 5(a) demonstrates that even in the extreme case of very low TX_{AP} without any interference some of the nodes (2 in this figure) are abnormal and suffer from low PDR.

The set of abnormal nodes remained small also when we increase TX_{AP} to higher bit-rates until 36 Mbps, as shown in Fig. 5(b) and 5(c). The number of abnormal nodes increases significantly once TX_{AP} reaches 48 Mbps. Surprisingly, the set of abnormal nodes are not the same in all experiments. This dynamic change of abnormal nodes is handled by the on-line rate adaptation mechanism in Section VII.

E. Feedback Node Selection

We now compare our FB node selection criterion with some other heuristics, and in the process, validate hypothesis $H4$.

FB Node Selection: We consider the following algorithms, including two flavors of AMuSe that select either the LQ or the PDR as the metric using which the AP selects FB nodes.

- 1) AMuSe(LQ) – AMuSe algorithm based on LQ.
- 2) AMuSe(PDR) – AMuSe algorithm based on PDR.
- 3) K-random – AP randomly selects a given number, k , of nodes as FB nodes.
- 4) K-best – AP selects k nodes with lowest LQ as FB nodes.

In addition, each non-FB node i is associated with a FB node, denoted as FB_i . The association in AMuSe occurs during cluster creation, while in other schemes the non-FB nodes are associated with the nearest FB node.

Evaluation Metrics: We define the *false positive ratio* of node i , denoted by FPR_i , as the fraction of packets not received by non-FB node i but has been reported as received correctly by its associated FB node among all the transmitted packets. A high false positive ratio implies that feedback from FB nodes is not effective. This is because non-FB nodes might not be receiving a high number of packets even if the FB node indicates otherwise.

A FB node is termed *satisfied* if its $PDR \geq H$ (for example, $H = 90\%$ in our experiments), since such FB nodes report satisfying multicast service⁹. A non-FB node i is termed

⁹The satisfaction threshold G (here chosen to be $G = 90\%$) can be modified depending on the multicast stream.

a *poorly represented node* if it has a high false positive ratio and is associated with a satisfied FB node. We characterize such a node i by $PDR_{FB_i} \geq 90\%$, and $FPR_i > 10\%$. In the ideal case, we want FPR_i to be very low for all nodes if possible, regardless of the state of their FB node, but the poorly represented nodes represent a violation of any multicast QoS that a provider wants to support, and therefore the number of such nodes should be as low as possible.

To ensure a fair comparison across all of the four schemes, we evaluate the number of poorly represented nodes for all schemes with the same number of FB nodes, denoted as f , and vary the number of FB nodes by changing D -adjacency values (i.e., cluster radius) for the AMuSe(LQ) and AMuSe(PDR) schemes, and by changing k to the corresponding values for the K-best and K-random schemes. We also evaluate the sensitivity of AMuSe to various values of D .

Experimental Results:

With no external noise, we set $TX_{AP} = 48$ Mbps, and plot the number of poorly represented nodes as f increases in Fig. 6(a) for the four schemes averaged over five runs of the experiment. The results for one run of K-random scheme are averages over 20 random feedback selections to ensure 95% confidence level. We see that AMuSe(PDR) does not have any poorly represented nodes at all for all values of f . AMuSe(LQ) is slightly better than K-best feedback selection, whereas all three schemes are better than K-random.

In Fig. 6(b), we repeat these experiments, but after injecting noise powers at varying levels for all feedback selection schemes. The number of feedback nodes f is fixed at 27, which is the typical number of FB nodes obtained by the AMuSe(PDR) scheme for $D = 3$ meters.

Again, AMuSe(PDR) outperforms all other schemes. AMuSe(LQ) performs poorly at high noise levels, which is intuitive since the LQ metric does not correlate well with PDR in the high-noise region. In Fig. 6(c), we study the effects of increasing the number of feedback nodes in case of high noise levels at -35 dBm¹⁰. We see that AMuSe(PDR) outperforms all other schemes.

¹⁰At noise powers higher than -35 dBm, significant number of nodes stop receiving any packets from the AP.

It is worthwhile to mention here that in case of two and four FB nodes, we see that AMuSe(PDR), AMuSe(LQ) and K-best schemes all result in zero poorly represented nodes. This is because the poorest performing nodes in the network are being selected, with $PDR < 90\%$. By definition these are abnormal nodes and consequently there are no satisfied FB nodes. This also gives us an intuition that selecting too few FB nodes will not be a good idea as in that case, the AP will be dictated by a set of worst performing abnormal nodes. For instance, Fig. 4(b) and 4(c) show that selecting only two or four FB nodes give FB nodes with $PDR < 30\%$. Clearly, these nodes do not represent the overall multicast service quality that most of the nodes experience.

In addition to not suffering from poorly represented nodes, the AMuSe(PDR) scheme has a *cumulative false positive ratio of less than 1%, summed over all nodes, across all the experiments*. These results validate the practicality the FB node selection criterion based on PDR values.

Location Error Sensitivity: We evaluate the sensitivity of AMuSe to errors in node location estimation by injecting errors into reported node locations. The errors are picked from a Gaussian distribution with $\mu = 0, \sigma = 7$ meters, with the D -adjacency parameter set to $D = 4.5$ meters. Despite this, there is no increase in the number of poorly represented nodes when AMuSe(PDR) uses these incorrect coordinates. We start observing a mild increase in the number of such nodes only when $\sigma \geq 10$ meters, and this levels off for higher location error. Having very high inaccuracy in position estimation results in AMuSe(PDR) degenerating to the K-best scheme. Obviously, this gives low false positive ratios, but the selected FB nodes may not represent the network well.

These results validate the practicality of the FB node selection criterion based on PDR values and demonstrate that AMuSe(PDR) is resilient to location error to a good degree.

VII. ON-LINE RATE ADAPTATION EXPERIMENTS

So far we have evaluated the effectiveness of AMuSe feedback selection using off-line analysis. In this section we use on-line experiments to demonstrate that coupled with a simple rate adaptation algorithm, the AMuSe feedback selection mechanism ensures high PDR for majority of the nodes¹¹.

A. Experimental Challenges

First, we discuss a number of experimental challenges we face.

A. PDR Variability: As expected in any wireless environment, PDR values in the nodes are time-varying even for a fixed AP multicast bit-rate. Such PDR variability impacts the stability of the FBN-LIST. We address this by defining a QoS hysteresis margin ($QoS_{HYST} = 3\%$) for PDR measurements at the nodes in order to prevent them from volunteering as FB nodes as a result of very small variations in their PDR measurements. QoS_{HYST} provides stability to the feedback

¹¹Note that we are not testing multicast source-rate adaptation, which can further improve performance, since it is orthogonal to the AMuSe system.

list, thus preventing FB node reconfiguration throughout the entire network.

B. Abnormal Nodes: As discussed in Section VI-D, even with TX_{AP} at low bit-rate and without external noise, abnormal nodes suffer from low PDR. For practical purposes, we are interested in excluding the small minority of nodes with abnormal QoS from the service in order to improve the overall network utilization while effectively serving the majority of the nodes. Abnormal nodes must not represent other nodes (except themselves) and they must be easily identifiable by the AP. In order to solve these two issues, we set the D -adjacency parameter to 0 for nodes with PDR value less than the given threshold H (i.e., abnormal nodes). Setting the D -adjacency parameter of a given node to 0 implies that its neighboring nodes will no longer consider the abnormal node as a valid FB node. In turn, this will cause the neighboring nodes to start sending VOLUNTEER announcements to become FB nodes¹². This setting of the D parameter guarantees that all the abnormal nodes (with PDR below $H - QoS_{HYST}$) volunteer to be FB nodes and the AP is informed about all the abnormal nodes, denoted by A .

C. Nodes Disconnecting from AP: High AP transmission bit-rates and overall interference in the system can lead to disconnection of some of the nodes on the grid, as seen in Fig. 4. High AP transmission bit-rates translate into more spectrum-efficiency, yet less robust modulations (16-QAM, 64-QAM) in the PHY layer. Analogously, low AP transmission bit-rates lead to less spectrum-efficiency, yet more robust modulations (BPSK, QPSK). These changes in the PHY layer have an impact on the transmission bit-rate of 802.11a beacon messages, which in turn result in some nodes losing association with the AP due to poor signal quality. In order to solve this issue, we have developed a modified version of the *ath5k* wireless driver that allows variable transmission bit-rate for multicast traffic and user data, while ensuring a fixed transmission bit-rate of 6 Mbps for AP beacons.

B. Rate Adaptation Algorithm

In our experiments, we set the *majority threshold* X to be 95%. The rate adaptation algorithm collects feedback information at the AP and attempts to update the transmission bit-rate periodically, to ensure that at least $X = 95\%$ of the nodes experience a PDR above $H = 90\%$ (in fact, AMuSe tolerants small PDR variability of $QoS_{HYST} = 3\%$). Nodes with $PDR \geq H$ are considered as normal nodes, whereas nodes with $PDR < H - QoS_{HYST}$ are flagged as abnormal nodes with D -adjacency value set to 0. Nodes with PDR values between $H - QoS_{HYST}$ and H keep their previous state.

Using a simple rate adaptation algorithm, AMuSe starts with low TX_{AP} bit-rate of 6 Mbps and iteratively increases or decreases the TX_{AP} bit-rate according to the PDR reports from normal and abnormal FB nodes. At each iteration, once

¹²Note that setting the D -adjacency parameter to 0 may lead to message bursts from FB node selection process. However, our experimental results show that this mechanism works well in practice without high message overhead.

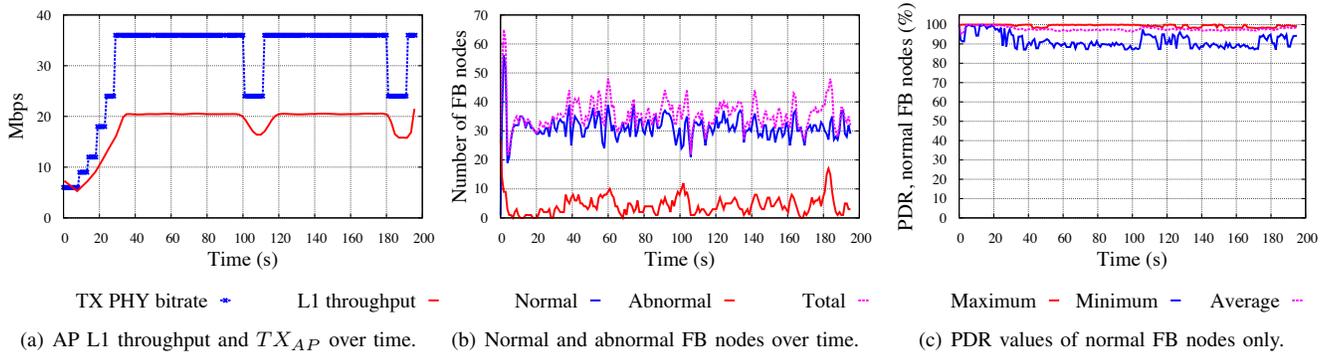


Fig. 7. Results from on-line rate adaptation experiments with $D = 3$ meters.

per second¹³, AMuSe performs the following operations:

(a) First, it calculates the number of abnormal nodes ($|A|$) and if $|A|$ is more than 5% of the receiver population the TX_{AP} bit-rate is reduced to the adjacent lower bit-rate supported by the standard.

(b) Otherwise, at least $X = 95\%$ of the nodes experience high PDR above $H - QoS_{HYST} = 87\%$ and AMuSe should decide whether to increase the bit rate to the next supported rate or keep the currently used TX_{AP} bit-rate. In order to avoid oscillations of the TX_{AP} bit-rate, the latter should be increased only if a large fraction of the nodes ($\geq X$) will continue to benefit from high PDR ($\geq H$) with the higher bit-rate. Since each increase of the TX_{AP} bit-rate causes reduction of the receiver PDR values, we expect the above condition to hold only if more than a fraction $X (= 95\%)$ of the node set experience very high PDR above some threshold $H_1 > H$, e.g., in our experiments we used $H_1 = 98\%$. Since the AP has PDR information only from the FB nodes, it estimates the number of receivers that experience very high PDR above H_1 and only if this set of receivers experiencing $PDR \geq H_1$ exceeds X , AMuSe increases the TX_{AP} bit-rate.

In order to allow nodes to adapt to a new AP transmission bit-rate, we enforce a delay of five iteration intervals (5 seconds) between two consecutive rate change actions and even longer delay of ten iteration intervals (10 seconds) between a rate decrease operation and a subsequent rate increase operation. These delays provide stability to the AP transmission bit-rate without significant impact on the system response time.

C. Experimental Setup

We set the PDR threshold $H = 90\%$ with $QoS_{HYST} = 3\%$ and we require that at least $X = 95\%$ of the nodes experience high PDR values of at least H . We run a set of experiments with D -adjacency value (D) of 3 meters to determine the FB cluster size and 230 nodes were involved in each experiment. Thus, the upper bound on the number $|A|$ of abnormal nodes is $\lceil 230 \times 5\% \rceil = 12$. We fix the experiment duration to 200 seconds in order to (a) allow the FB selection process to

converge to a stable set of FB nodes, and (b) show changes in the rate adaptation algorithm as well as convergence towards a high AP transmission bit-rate. Note that we only present results without any external interference.

D. Performance Indicators

In order to evaluate the system performance we use the following metrics:

(a) TX_{AP} bit-rate and layer 1 (L1) throughput at AP, which indicate the operating region of the rate adaptation algorithm.

(b) Number of FB nodes at any given time, both normal and abnormal.

(c) PDR measurements, including the minimum, maximum, average PDR measurements of the normal FB nodes in the FBN-LIST message.

E. On-line Experimental Results

Fig. 7 shows the evolution of the three performance indicators over time for $D = 3$ meters.

In Fig. 7(a), we see the variation of the AP transmission bit-rate and its effect on the L1 throughput. The system adequately converges to TX_{AP} bit-rate of 36 Mbps as expected by our analysis in Section VI-E, except for the occasional bit-rate drop to 24 Mbps.

Fig. 7(b) tracks the number of normal and abnormal nodes in the feedback list and explains the evolution of AP transmission bit-rate as seen in Fig. 7(a). For low TX_{AP} bit-rate, the number of abnormal nodes is very small, and it increases slightly and fluctuates between 0-10 for TX_{AP} bit-rate of 36 Mbps. Note that each drop of the TX_{AP} bit-rate is clearly correlated with a sporadic increase of the abnormal node set to above 12 nodes, for example, when the elapsed time is 100 and 180 seconds.

Fig. 7(c) shows that the PDR values, reported by normal FB nodes, stay within the acceptable limits given by $H - QoS_{HYST} = 87\%$.

F. Summary of Online Experiments

We have demonstrated that the use of the PDR metric combined with the AMuSe FB node selection algorithm can result in representative feedback that is of high quality to the AP. Furthermore, even with a simple rate adaptation mechanism on the AP, we were able to drive the multicast

¹³The iteration interval of the rate adaptation algorithm directly impacts system stability. The determination of the optimal interval as well as other details of the rate adaptation algorithm are beyond the scope of this paper.

stream to high bit-rates for most nodes while adapting to channel conditions if necessary. The problem of finding a proper balance between abnormal nodes identification, AP L1 throughput, and FB node selection stability is the aim of our future work.

VIII. CONCLUSION AND FUTURE WORK

In this paper, we presented the design and large-scale experimental evaluation of the AMuSe system for providing scalable and efficient multicast services for a large group of users in a small geographical region. AMuSe can be implemented as an application layer protocol on existing WiFi devices, and can provide an accurate and representative feedback on the quality of the multicast streams back to the multicast source. Our solution allows practical multicast streaming services to become a reality.

Our ongoing work spans the following directions: (a) fine tuning of control knobs in AMuSe, such as reporting frequency τ_{AP} and τ_{FB} , the node adjacency distance D , the abnormal node threshold H , and the majority threshold X , in order to achieve the balance of reporting overhead, operation stability and system throughput; (b) improving the rate adaptation mechanism for providing fast response to interference without compromising the system stability; (c) extending and testing the system ability to support multiple multicast streams; and (d) studying the capacity of AMuSe under a combined unicast-multicast model.

IX. ACKNOWLEDGMENTS

The authors would like to express their gratitude to Ivan Seskar from WINLAB (Rutgers University) for his critical support in conducting experiments and several useful technical discussions. We would also like to thank Josiah Hutchinson for helping with data processing and Dan Bienstock for providing access to his cluster. This work was supported by (while one of the authors was serving at) the National Science Foundation, and in part by NSF grant CNS-10-54856, NSF CIAN ERC under grant EEC-0812072, by the Spanish Ministry of Economy and Competitiveness under grant TEC2011-29700-C02-01, the Spanish Ministry of Education under grant FPU AP2009-5000, and the Generalitat de Catalunya under grant 2009-SGR-940.

X. DISCLAIMER

Any opinion, findings and conclusions or recommendations expressed here do not necessarily reflect the views of the National Science Foundation.

REFERENCES

- [1] "IEEE draft standard for information technology telecommunications and information exchange between systems local and metropolitan area networks - specific requirements, part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications - amendment: MAC enhancements for robust audio video streaming," IEEE, July 2011.
- [2] Y. Tanigawa, K. Yasukawa, and K. Yamaoka, "Transparent unicast translation to improve quality of multicast over wireless LAN," in *Proc. IEEE CCNC'10*, 2010.
- [3] "Cisco, white-paper, cisco connected stadium wi-fi solution," 2011. [Online]. Available: http://www.cisco.com/web/strategy/docs/sports/c78-675064_svcs.pdf
- [4] "Yinzcam," <http://www.yinzcam.com/>.
- [5] K. Pelechrinis, T. Salonidis, H. Lundgren, and N. Vaidya, "Experimental characterization of 802.11n link quality at high rates," in *Proc. ACM WINTECH'10*, 2010.
- [6] N. Hajlaoui and I. Jabri, "On the performance of IEEE 802.11n protocol," in *Proc. ACM WINTECH'12*, 2012.
- [7] X. Zhang, X. Peng, S. Fowler, and D. Wu, "Robust H.264/AVC video transmission using data partitioning and unequal loss protection," in *Proc. IEEE CIT'10*, 2010.
- [8] D. Aguayo, J. Bicket, S. Biswas, G. Judd, and R. Morris, "Link-level measurements from an 802.11b mesh network," *Proc. ACM SIGCOMM'04*, 2004.
- [9] M. V. Marathe, H. Breu, H. B. Hunt III, S. S. Ravi, and D. J. Rosenkrantz, "Simple heuristics for unit disk graphs," *Networks*, vol. 25, pp. 59–68, 1995. [Online]. Available: citeseer.nj.nec.com/marathe95simple.html
- [10] "ORBIT testbed," <http://orbit-lab.org/>.
- [11] R. Chandra, S. Karanth, T. Moscibroda, V. Navda, J. Padhye, R. Ramjee, and L. Ravindranath, "DirCast: a practical and efficient Wi-Fi multicast system," in *Proc. IEEE ICNP'09*, 2009.
- [12] N. Choi, Y. Seok, T. Kwon, and Y. Choi, "Leader-based multicast service in IEEE 802.11v networks," in *Proc. IEEE CCNC'10*, 2010.
- [13] J. K. Kuri and S. Kumar, "Reliable multicast in multi-access wireless LANs," *ACM/Kluwer Wirel. Netw.*, vol. 7, pp. 359–369, 2001.
- [14] M.-T. Sun, L. Huang, A. Arora, and T.-H. Lai, "Reliable MAC layer multicast in IEEE 802.11 wireless networks," in *Proc. IEEE ICPP'02*, 2002.
- [15] X. Wang, L. Wang, Y. Wang, Y. Zhang, and A. Yamada, "Supporting MAC layer multicast in IEEE 802.11n: Issues and solutions," in *Proc. IEEE WCNC'09*, 2009.
- [16] O. Alay, T. Korakis, Y. Wang, and S. Panwar, "Dynamic rate and FEC adaptation for video multicast in multi-rate wireless networks," *ACM/Springer Mobile Netw. and Appl.*, vol. 15, no. 3, pp. 425–434, 2010.
- [17] H.-T. Chiao, S.-Y. Chang, K.-M. Li, Y.-T. Kuo, and M.-C. Tseng, "WiFi multicast streaming using AL-FEC inside the trains of high-speed rails," in *Proc. IEEE BMSB'12*, 2012.
- [18] M. Wu, S. Makharia, H. Liu, D. Li, and S. Mathur, "IPTV multicast over wireless LAN using merged hybrid ARQ with staggered adaptive FEC," *IEEE Trans. Broadcast.*, vol. 55, no. 2, pp. 363–374, 2009.
- [19] W.-S. Lim, D.-W. Kim, and Y.-J. Suh, "Design of efficient multicast protocol for IEEE 802.11n WLANs and cross-layer optimization for scalable video streaming," *IEEE Trans. Mobile Comput.*, vol. 11, no. 5, pp. 780–792, 2012.
- [20] Z. Feng, G. Wen, C. Yin, and H. Liu, "Video stream groupcast optimization in WLAN," in *Proc. IEEE ITA'10*, 2010.
- [21] S. Sen, N. K. Madabhushi, and S. Banerjee, "Scalable WiFi media delivery through adaptive broadcasts," in *Proc. USENIX NSDI'10*, 2010.
- [22] Y. Park, C. Jo, S. Yun, and H. Kim, "Multi-room IPTV delivery through pseudo-broadcast over IEEE 802.11 links," in *Proc. IEEE VTC'10*, 2010.
- [23] Z. Li and T. Herfet, "HLBP: a hybrid leader based protocol for MAC layer multicast error control in wireless LANs," in *Proc. IEEE GLOBECOM'08*, 2008.
- [24] J. Yoon, H. Zhang, S. Banerjee, and S. Rangarajan, "MuVi: A multicast video delivery scheme for 4G cellular networks," in *Proc. ACM MOBICOM'12*, 2012.
- [25] P. Mirowski, H. Steck, P. Whiting, R. Palaniappan, M. MacDonald, and T. K. Ho, "KL-divergence kernel regression for non gaussian fingerprint based localization," in *Proc. IPIN'11*, 2011.
- [26] H. Rahul, F. Edalat, D. Katabi, and C. G. Sodini, "Frequency-aware rate adaptation and MAC protocols," in *Proc. ACM MOBICOM'09*, 2009.
- [27] A. Vlavianos, L. Law, I. Broustis, S. Krishnamurthy, and M. Faloutsos, "Assessing link quality in IEEE 802.11 wireless networks: Which is the right metric?" in *Proc. IEEE PIMRC'08*, 2008.
- [28] S. Kaul, M. Gruteser, and I. Seskar, "Creating wireless multi-hop topologies on space-constrained indoor testbeds through noise injection," in *Proc. IEEE TRIDENTCOM'06*, 2006.