Light-weight Feedback Mechanism for WiFi Multicast to Very Large Groups - Experimental Evaluation

Varun Gupta, Yigal Bejerano, Craig Gutterman, Jaime Ferragut, Katherine Guo, Thyaga Nandagopal, and Gil Zussman

***Abstract*—WiFi Networks have been globally deployed and most 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. In this paper, we present the AMuSe system, whose objective is to enable scalable and adaptive WiFi multicast services. AMuSe is based on accurate receiver feedback and incurs a small control overhead. Specifically, we develop an algorithm for dynamic selection of a subset of the multicast receivers as feedback nodes which periodically send information about the channel quality to the multicast sender. This feedback information can be used by the multicast sender to optimize multicast service quality, e.g., by dynamically adjusting transmission bitrate. AMuSe does not require any changes to the standards or any modifications to the WiFi devices. *We implemented AMuSe on the ORBIT testbed and evaluated its performance in large groups with approximately 200 WiFi devices*, both with and without interference sources. Our extensive experiments demonstrate that AMuSe can provide accurate feedback in a dense multicast environment. It outperforms several alternatives even in the case of external interference and changing network conditions.**

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

I. Introduction

Recent years have witnessed a rapid growth of mobile devices equipped with an IEEE 802.11 (WiFi) interface [3] which allow users to access the Internet anywhere and any time. Yet, due to a combination of high bandwidth requirements and a shortage of wireless spectrum, it is complex to serve rich multimedia content (such as video streams) to users clustered in crowded areas. The growing need to support larger demands for multimedia content using limited resources in dense areas has prompted the design of several solutions by both industry and academia. Many of these solutions [4]–[6] are typically based on dense deployments 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 APs.

Current state of the art techniques using IEEE 802.11 for content delivery leverage either unicast or multicast data delivery. Commercial products [5], [6] rely on unicast for streaming the content to individual users. With standards such as 802.11ac promising total speeds up to 800 Mbps using multi-user MIMO, it is theoretically possible to serve video streams to hundreds of users. However, recent studies [7], [8] throw cold water on this promise. A large number of neighboring APs leads to hidden terminal problems and this coupled with increased interference sensitivity due to channel bonding, makes the entire approach highly susceptible to interference. Extrapolating from studies on 802.11n [7], [8], it seems that 802.11ac-based unicast to multiple receivers may not be able to support more than a hundred users, assuming all of them have 802.11ac capable devices.

On the other hand, WiFi multicast services are rarely used by practical content delivery applications. Standard WiFi broadcast/multicast frames are transmitted at a fixed and low bitrate without any feedback. This raises several known reliability and efficiency issues. While some commercial products [5] are experimenting with WiFi multicast deployments for crowded environments, there remain several challenges to its widespread adoption. In particular, a recently published IETF Internet Draft highlights several open technical problems for WiFi multicast [9]. High packet loss due to interference and the hidden node problem can significantly degrade service quality. On the other hand, transmitting at low bitrates leads to low network utilization. As described in Section II, there are numerous studies that propose solutions for overcoming these limitations from two aspects. One aims to reduce the overhead of feedback information to the multicast sender. The other aims to improve message reliability based on available feedback information. All the existing schemes, however, suffer from one or more issues including lack of scalability, inability to guarantee high service quality, or compliance with existing standards. Further, *none of the schemes have been tested experimentally at scale.*

A. The AMuSe System

We consider the use of WiFi multicast to address the challenge of providing scalable and efficient delivery of multimedia content to a very large number of WiFi nodes in a

V. Gupta, C. Gutterman, and G. Zussman are with the Department of Electrical Engineering, Columbia University, NY, USA (email: varun@ee.columbia.edu, clg2168@columbia.edu, gil@ee.columbia.edu).

Y. Bejerano and K. Guo are with Bell Labs, Nokia, Murray Hill, NJ, USA (email: bej@research.bell-labs.com, kguo@bell-labs.com).

J. Ferragut is with the Architecture and Technology Group, ARM Ltd., Cambridge, UK (e-mail: jaime.ferragut@arm.com).

T. Nandagopal is with the National Science Foundation, Arlington, VA, USA (email: thyaga.n@gmail.com).

Preliminary versions of this paper were presented in IEEE ICNP'13 [1] and GENI Research and Educational Experiment Workshop'14 (invited) [2].

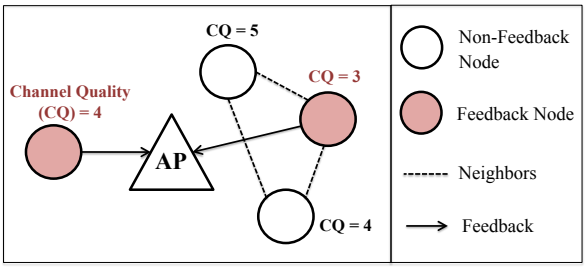


Fig. 1. Feedback node selection by *AMuSe*. A node with the poorest channel quality in every neighborhood is selected as a Feedback node. Each feedback node periodically sends updates about the service quality to the Access Point.

small geographical region (e.g., sport arenas, lecture halls, and transportation hubs). This is an attractive approach for delivering live video content to a dense user population that shares common interests (e.g., providing simultaneous video feeds of multiple camera angles in a sports arena).

The core challenge in providing such a service is collecting limited yet sufficient feedback from the users for optimizing the network performance. To address this challenge, *we introduce AMuSe (**A**daptive **Mu**lticast **Se**rvices), a low-overhead feedback system which leverages the existing WiFi standards for tuning the network parameters, i.e., optimizing the network utilization while preserving Quality of Service (QoS) requirements.* AMuSe is based on the following hypothesis, which was reported in [10] and is validated in this paper.

Main Hypothesis: A cluster of adjacent nodes experience similar channel quality and suffer from similar interference levels. Hence, a node v with a worse channel condition than its adjacent neighbors can represent the service quality observed by the nodes in the cluster.

AMuSe dynamically divides the nodes in a network into clusters based on the adjacency of nodes and maximum cluster size (D m). In each cluster, one node is selected as a *Feedback (FB) node* and the FB node updates the AP about its service quality, e.g., channel quality (an example is shown in Fig. 1). The AP, in response, may take several actions such as[1]:

(i) **Rate Adaptation:** AMuSe can allow the APs to transmit multicast traffic at the highest possible bitrate while meeting constraints set by a network operator, i.e. ensuring high *Packet Delivery Ratio* (PDR) for a large fraction of the nodes. Our recent work on multicast rate-adaption is given in [11].

(ii) **Tuning FEC:** We demonstrate in this paper that ensuring 100% packet deliveries to all nodes is challenging. In large multicast groups, even a small amount of packet losses at nodes could lead to large packet retransmissions. In such situations, dynamically tuning application-level FEC might be a more suitable option. Feedback from AMuSe can be used to adjust the amount of FEC dynamically.

(iii) **Detecting Interference:** AMuSe collects detailed packet statistics which can be used to identify causes of packet loss in the network such as collisions and noise. For instance, packet losses that occur at the same time at multiple nodes can help pinpoint the location of the interference.

AMuSe can be implemented as a light-weight application on any WiFi enabled device with minor or no modifications to the receiver devices and does not require changes to the existing 802.11 standard. The AMuSe system allows multicast service operators to balance between the number of FB nodes, the accuracy of the feedback, and the system convergence time by controlling AMuSe parameters, such as the cluster radius D. AMuSe ensures that every node is at most D m away from an FB node with similar or weaker channel quality. To ensure sparse FB node density, any pair of FB nodes are at least D m apart which results in low communication overhead. The problem of selecting FB nodes which meet the above requirements is a variant of the well known *Minimal Independent Dominating Set* problem [12]. Although this problem is NP-hard, we prove that AMuSe can find a solution with a small constant approximation ratio.

B. Experimental Evaluation

We evaluated AMuSe on the large-scale ORBIT testbed [13] using over 200 WiFi nodes by implementing AMuSe on the application layer at each device. In all of our experiments, one node served as the AP and it sent a continuous multicast flow to all the other nodes, which acted as receivers. We first study the variation of channel quality metrics in different scenarios, (e.g., varying external interference levels, different transmission bit rates). The observations from these experiments serve as guiding principles for the design of AMuSe.

We observe that during any experiment, some nodes, which will be defined as *abnormal nodes*, suffer from low PDR, even when the AP is transmitting at a low bitrate and there is no external interference. Furthermore, this set of abnormal nodes varies across experiments.

We collected detailed channel and service statistics from all the nodes. They include the *Link Quality*[2] (LQ) reported by each node's WiFi card as representative of its observed received signal strength (RSS), its PDR, and its distance from the AP. Our preliminary evaluations show only moderate correlation between the nodes' LQ and the experienced PDR and a weak correlation between the nodes' distance from the multicast AP and the PDR values.

To validate the Main Hypothesis, we consider all the possible clusters with radius 3 and 6 m and calculate the Standard Deviation (STD) of the LQ and PDR values in the clusters at different bitrate and noise-levels. Our experiments indeed show low LQ and PDR STDs between the nodes in a cluster. However, as we increase the transmission bitrate or the noise level, we observe an increase in STD for the PDR values. We also notice that clusters with a small radius have lower LQ and PDR STDs than larger clusters.

We assess the feedback reports produced by AMuSe when the channel quality is evaluated according to the nodes' LQ, PDR, or a combination of them. These variants are denoted as AMuSe-LQ, AMuSe-PDR, and AMuSe-Mix respectively. We compare their performance to other feedback node selection schemes; *K-Worst* [14], [15], which selects the receivers with the worst channel condition as FB nodes, and *Random*, which

[1]The actions of the AP will require changes only at the AP side which is relatively straightforward.

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

selects a fixed number of random FB nodes. To evaluate the quality of an FB node selection, we compute the number of non-FB nodes that experience PDR value strictly lower than their respective FB node. We refer to these nodes as *Poorly Represented Nodes* (PRNs). We show that AMuSe-PDR and AMuSe-Mix produce a negligible number of PRNs and they outperform the other schemes when evaluated with different multicast bitrates and various noise levels. AMuSe-LQ and K-Worst have comparable performance, and are significantly better than the Random scheme.

Furthermore, we assess the performance of AMuSe as a service quality predictor in the event of environment changes. More specifically, we first select the FB nodes of the different variants at a given network setting. We then, compute the number of poorly represented nodes when using the same FB nodes, but after changing the multicast bitrate or the noise-level. We observe that at low bitrates AMuSe-LQ has slightly less PRNs than AMuSe-PDR, while AMuSe-PDR has similar performance to K-Worst. We notice a different trend when operating at a high multicast bitrate, in which AMuSe-PDR outperformed AMuSe-LQ and K-Worst. In all evaluations AMuSe-Mix was the best variant while Random, suffered from a very high number of PRNs. We explain these observations and provide additional results in Section VI.

Our experimental results demonstrate the ability of AMuSe to effectively provide feedback about the performance and quality of wireless multicast services. In turn, this feedback can be used for tuning the network parameters (e.g., rate adaptation, FEC configuration, and interference classification) to optimize multimedia content delivery.

C. Paper Organization

We describe the network settings and our objectives in Sections III and IV respectively. We present testbed evaluation of the design of AMuSe in Section V and the experimental results of evaluating channel quality metrics in Section VI. Finally, the evaluation of the performance of AMuSe is presented in Section VII for both the static and dynamic cases before concluding in Section VIII.

II. Related work

Various methods have been proposed for multimedia content dissemination to multiple receivers. They leverage either unicast or multicast data delivery. This brief overview describes the most relevant studies to our paper (comprehensive overview on wireless multicast appears in [16]). Commercial products [5], [6] rely on unicast for streaming content to individual users. This approach requires deployment of numerous APs and it does not scale to crowded areas. Alternatively, the basic 802.11 multicast mechanism without any node feedback simply sets the transmission bitrate to the lowest rate. Cellular networks also operate without any node feedback and set the transmission bitrate to a low value, assuming some nodes are located near the cell edge. Any multicast mechanism without feedback results in low network utilization.

Many of the schemes to improve multicast services are based on integrating Automatic Repeat Request (ARQ) mechanisms into the protocol architecture [14], [15], [17]–[19], adding Forward Error Correction (FEC) packets to the multicast stream [20], [21], or both [22]. Other studies propose rate adaptation mechanisms for improved network utilization [23].

In all cases, a key requirement is having appropriate feedback from the receivers regarding their observed service quality. These feedback mechanisms can be classified as follows: (i) *Individual Feedback* from multicast receivers, (ii) Leader-Based Protocol with acknowledgements (LBP-ACK), (iii) *Pseudo-Broadcast*, and (iv) *Leader-Based Protocol* with negative acknowledgements (LBP-NACK).

Individual Feedback mechanisms require all receivers to send acknowledgements of received packets either at the link layer [14], [19], [24]–[26], the application layer [22], or using periodic updates [20], [27]. With *More Reliable Groupcast* (MRG) [3], [28] 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], [29] provides scalability by selecting a subset of the receivers to provide feedback. The Pseudo-Broadcast approach [15], [17], [27], [30], 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 [18], [23], [31] 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.

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. 2(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. 2(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 scheme requires changes to the standard and suffers from lack of reliability since a non-leader cannot reply with a NACK if it cannot identify a corrupted packet. Furthermore, due to the capture effect, the AP may be able to decode the ACK and ignore NACK messages. A major drawback of the LBP-NACK scheme is lack of fine-grained information about packet losses. Consider an example with 100 nodes in a multicast group, each with PDR of 99%. The expected fraction of packets for which NACK messages are received is $1 - .99^{100}$, which translates to roughly 63% of the packets. Thus, even in the case of network performing well, the AP observes poor performance.

Table I summarizes the main features of existing approaches. In summary, at least one of the following weaknesses hinders their performance: (i) requirement of feedback from a large number of receivers, (ii) ignorance of AP to interference-related packet loss, (ii) low network utilization to compensate

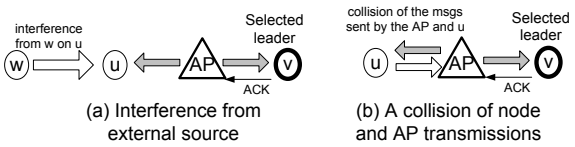


Fig. 2. Unreliable packet delivery by the LBP and the Pseudo-Broadcast approach.

TABLE I
MULTICAST: FEATURES OF RELATED WORK

	Scalable (a)	**QoS Guarantees (b)**	**High Util. (c)**	**Standards Compatible (d)**	**Low Cost (e)**
Unicast	x	√	x	√	x
Basic multicast	√	x	x	√	√
Individual Feedback	x	√	x	x	√
Pseudo Broadcast	√	x	x	√	√
LBP-NACK	√	x	x	x	√
AMuSe	√	√	√	√	√

for lack of feedback information or due to abnormal nodes, (iv) requirement of changes to standard WiFi protocol, or (v) expensive deployment of numerous APs. This motivates our desire for a scalable solution that improves reliability of multimedia delivery for WiFi deployments.

III. NETWORK SETTING

We consider an IEEE 802.11 WLAN and focus on a single AP serving a dense deployment of WiFi devices or *nodes*. A multicast server sends data to the AP and the AP transmits this data using multicast to all the nodes in its transmission range. There could be several sources of external interference in the network including transmissions from nodes within the network, adjacent APs, and nodes outside the network.

We follow the model where a node may report its service quality (e.g., channel quality) to an AP or multicast server. The AP or the multicast server, in response, may decide to adjust the FEC, adjust the transmission bitrate, retransmit lost packets, or execute a combination of the above. In practice, the AP and the multicast server are two separate logical entities and may reside in multiple network layers. Only the AP, however, is responsible for adjusting the network layer parameters. To simplify presentation, in the rest of the paper we refer to AP as a representation of the combination of an AP and a multicast server.

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 [32]) with an accuracy of a few meters, and also determine if they are *static*[3] or *mobile*.

IV. OBJECTIVE

We focus on designing a light-weight feedback system for supporting scalable WiFi multicast services for a very large number of nodes. This system allows APs[4] to monitor the network conditions and to take appropriate actions for improving the multicast service quality while meeting various service delivery constraints. We rely on the following observation reported in [10]:

Observation: *A cluster of adjacent nodes experience similar channel quality and suffer from similar interference levels. Hence, a node v with worse channel condition than its adjacent neighbors can represent the service quality observed by the nodes in the cluster.*

Based on this observation, the nodes can be grouped into clusters of adjacent nodes and a single *Feedback (FB) node* from each cluster can represent that particular cluster. The FB node can be used to report the channel quality of the cluster to the AP. Our feedback system should ensure the following requirements:

(i) The FB nodes should accurately represent the network conditions in their neighborhood. This implies that the channel state experienced by non-FB nodes should not be significantly worse than the channel state reported by FB nodes.

(ii) The FB nodes should be well distributed throughout the network. In other words, the distance between the FB and non-FB nodes should be small. This ensures that the AP is informed about any interference even if it affects a small area.

(iii) The FB nodes should be responsive to changes of the service condition and should accurately report the impact of environmental changes, such as the multicast bitrate or external interference.

We now provide a formal definition of our objective. Given any FB node selection scheme and assume that every non-FB-node is represented by a single FB-node, typically the closest FB-node. A non-FB-node is considered a *Poorly Represented Node* (PRN) if its PDR is $\epsilon > 0$ below the PDR of its representing FB-node. We refer to ϵ as the *PRN Gap*. Consequently, our objective can be defined as follows;

Objective: *Consider an upper bound on the number of FB nodes or their density[5] as well as a fixed PRN-Gap $\epsilon > 0$. Design a low-communication FB node selection system that minimizes the following metrics:*

- *Number of PRNs in normal operation as well as after environment changes, e.g. bitrate or noise level changes.*
- *Maximum distance between a non-FB-node and its representing FB node.*

V. THE AMUSE SYSTEM

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

[3]We consider a node static, if its movement is restricted to a few meters.

[4]To simplify our presentation, we assume that AMuSe is implemented as a software module on the APs. In practice, AMuSe can be realized as an independent server or even a cloud service.

[5]The FB node density can be enforced by requiring a minimal distance D between any two FB nodes.

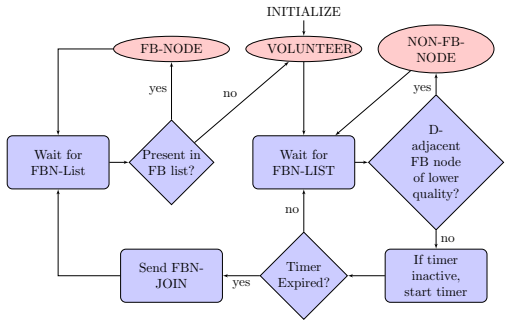


Fig. 3. State diagram of the AMuSe FB node selection algorithm at each node. All nodes initialize in the VOLUNTEER state.

(i) Each node should be D-adjacent to an FB node.
(ii) An FB node must have similar or weaker channel quality than its D-adjacent nodes.
(iii) Any two FB nodes cannot be D-adjacent.

In order to evaluate the channel quality, various metrics can be considered, including *Received Signal Strength* (RSS), *Signal-to-Noise Ratio* (SNR) and *Packet Delivery Ratio* (PDR). We experimentally compare LQ^2 and PDR as channel quality metrics in Section VI.

A. The Feedback Node Selection Algorithm

We present a semi-distributed process for FB node selection, where some nodes volunteer to serve as FB nodes, and the AP selects the best candidates. If node location information and observed channel quality are known, then the AP can easily select the ideal set of FB nodes. Yet, this is not feasible in practice for large groups. Hence, 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 that a small set of FB nodes meeting the above requirements is selected.

The AP periodically (e.g., once every $\tau_{AP} = 500$ ms in our experiments) multicasts an FBN-LIST message with a list of FB nodes (these messages can be sent multiple times for reliable transmissions and do not incur overhead, since they are 1-2 packets long). Each entry in the FBN-LIST contains the node ID[6], its reported location[7], its reported channel quality, and a measure of the PDR[8].
Each node is in one of three states:

- **FB-NODE -** A node that has been selected as FB node.
- **VOLUNTEER -** A node that is not aware of any D-adjacent FB node with lower or similar channel quality and can serve as an FB node.
- **NON-FB-NODE -** A node that either is in a transient state or is aware of a D-adjacent FB node with similar or lower channel quality.

Fig. 3 presents the state transition diagram for each node. When a node v joins the network, 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). The random timer solves the problem of many nodes overwhelming the WiFi channel and AP with FBN-JOIN messages in the situation of changes in channel condition. 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 sends 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 the observed channel quality (e.g., the node PDR and LQ). 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 an FBN-LIST message, if an FB node v does not find itself on the FBN-LIST, it ceases to be in the FB-NODE state. In this case, the node returns to the VOLUNTEER state and waits for the next FBN-LIST to either (i) switch to the NON-FB-NODE state due to the existence of a D-adjacent node of lower quality, or (ii) send the FBN-JOIN message again after the back-off timer expires.

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

Each FB node periodically (e.g., once every τ_{FB} = 500 ms in our experiments) sends REPORT messages to update the AP about the channel and service quality experienced by the node, and thus its representative cluster. 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.

A few aspects of the AMuSe system are worth pointing out.

(i) AMuSe does 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.
(ii) AMuSe does 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 [33].
(iii) AMuSe provides variable levels of reliability by fine-tuning the combination of AP node selection frequency τ_{AP}, the receiver reporting frequency, τ_{FB}, the maximum receiver back-off timer T, and the node adjacency distance D. AMuSe can ensure more reliable and frequent reports at a cost of more overhead. Instead of a single

[6]Nodes can be assigned temporary virtual IDs to maintain privacy.
[7]Relying on a user to be truthful about its location/channel quality could lead to denial-of-service attacks. Yet, we shelve this orthogonal discussion.
[8]This can be easily changed to report the last acknowledged packet sequence number to support finer granularity of message reliability.

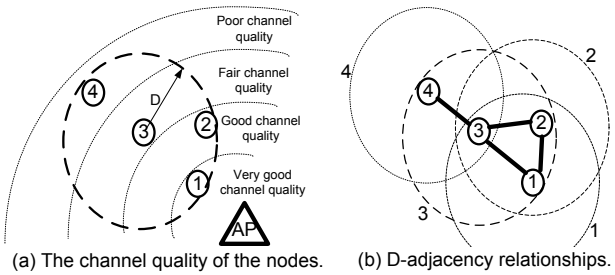


Fig. 4. An example of a wireless network a single AP and 4 receivers. All 3 requirements described in Section V for an accurate feedback selection are important for this example.

control, AMuSe provides multiple control knobs, giving greater flexibility to the operator to provide different types of service for various multicast streams.

(iv) Fourth, as described above and discussed in Section VIII, AMuSe reports can be used for optimizing different aspects of WiFi multicast services, such as rate-adaptation, FEC configuration and interference classification. To this end, the REPORT messages may carry different information. For instance, in [1] we showed that PDR and LQ information is sufficient for performing rate adaptation, while reporting about received and lost packets is required for interference classification.

B. Illustrative Example

Consider the network shown in Fig. 4(a) with a single AP and four receivers. Assume that numbers labeling the nodes denote their 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. 4(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 on the quality and density of the set of FB nodes. Assume first that the FB nodes have to meet only requirement (i) and (ii), but not (iii). 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 an FB node. At the end of the process, the network contains four FB nodes. It is easy to see that this approach does not scale for large groups.

Now, let us assume that requirement (iii) is enforced. Right after a node joins the network, 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 (iii). After node 3 joins, it becomes an 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 an FB node, while node 3 becomes a non-FB node. In addition, node 2 becomes an FB node again. Notice that node 2 switches state twice, after node 3 and 4 joins respectively. However, after each node joins the multicast group, the set of FB nodes is optimal.

This example shows that while AMuSe FB node selection algorithm satisfies all three requirements, it may cause churn as nodes enter and leave the FB-NODE state. We show next that the selected set of FB nodes is near-optimal when the set of nodes receiving the multicast do not change.

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. The list of candidates at the AP contains the current FB nodes as well as the nodes in the VOLUNTEER state. Thus, the AP is responsible to trim unnecessary candidates to select a small set of FB nodes such that any pair of nodes in the set are not D-adjacent.

The problem of finding the minimum set of FB nodes that meets the three requirements above is a variant of the *minimum dominating set problem*, which is a known NP-complete problem even in the case of unit disk graph [12]. Below we present a heuristic algorithm that selects a 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 an FB node and remove v and all its D-adjacent nodes from L. The algorithm ends when L is empty.

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

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

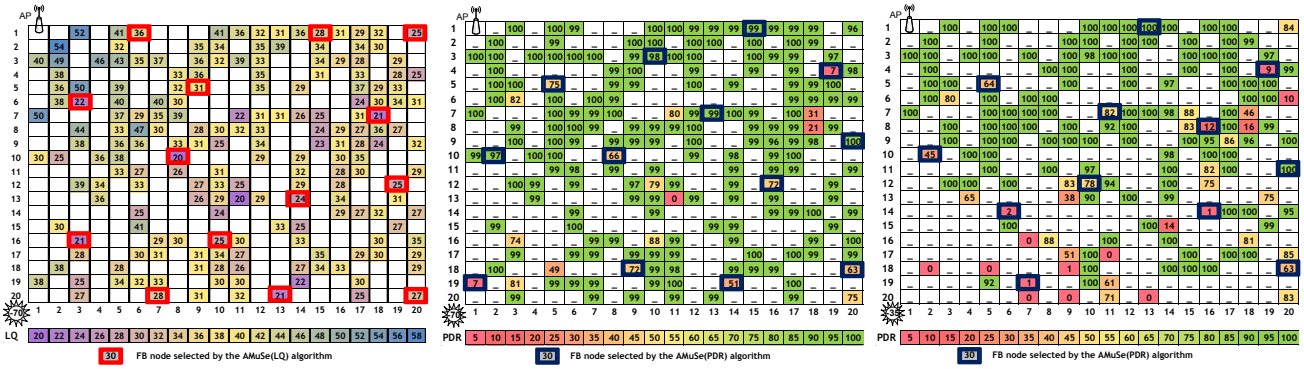
For proof see supplementary materials.

Stability vs. optimality trade-off: As illustrated in Section V-B, a naive implementation of the heuristic algorithm 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. In our experiments, we also observed rapid switching of FB nodes due to minor variations in channel qualities. In this case, ensuring that the difference between channel quality of a non-FB and FB node is greater than some value greater than zero before a non-FB node volunteers is an effective solution. Although striking a proper balance between system stability and optimality of the FB node selection is a central topic in the design of *AMuSe*, it is beyond the scope of this paper.

VI. Experimental Evaluation of Testbed Environment

We validated AMuSe experimentally using the 400-node ORBIT testbed [13]. We describe these experiments in this section. We use the *Link Quality*[2] (LQ) metric reported by a node's WiFi card as representative of its observed RSS. We first consider the following set of auxiliary hypotheses used to validate our main hypotheses in Section I.

- **H1:** There is a correlation between the PDR and LQ values observed by a node.
- **H2:** Clustered nodes experience similar LQ and PDR.
- **H3:** Clustered nodes suffer from similar interference.



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

(b) Packet Delivery Ratio Heatmap, noise = -70 dBm.

(c) Packet Delivery Ratio Heatmap, noise = -35 dBm.

Fig. 5. Link Quality (LQ) and Packet Delivery Ratio (PDR) heatmaps at the AP for $D = 6$ meters with transmission bitrate 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. 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. Note that the minimum threshold below which a node does not become an FB node is configurable.

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.

A. The ORBIT Testbed and Experiment Settings

The ORBIT testbed [13] consists of a dynamically configurable grid of 20×20 (400 overall) nodes each with an 802.11 radio. The grid separation between nodes is 1 meter and in addition, the testbed provides a noise generator with four 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 (nodes with hardware issues) in the grid before every experiment. This results in approximately 200 nodes participating in each experiment.

We implemented the AMuSe system as an application layer program for the AP and the clients, running on all nodes. Each node is identified by its (row, column) location. The node at the corner $(1, 1)$ serves as a single multicast AP, configured in master mode, and it uses channel 40 of 802.11a[9] 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 *at most a quarter of the transmission range of an AP*. Each UDP packet is 1400 bytes in payload length and the payload data contains sequence number for each packet in order to identify missing packets at the nodes. While we consider a single multicast group in our experiments, AMuSe can allow for monitoring of several multicast groups individually. If several multicast groups should be monitored together, then a control multicast group can be setup.

[9]We observed that channel 40 at the 5 Ghz band suffers from lower external interference levels on the ORBIT grid than the channels at 2.4Ghz band.

Every node keeps track of the parameters described in Table II, which we process off-line after each experiment. The received or dropped packets are marked by 1s or 0s respectively in a boolean vector P_i^{vec} stored at each node i. The *packet delivery ratio* (PDR) value of each node i is calculated from its P_i^{vec} vector. Note that the throughput measured at each node is a function of the PDR as well as the bitrate and is different from the transmission throughput at the AP. The testbed hardware and software allows us to measure the LQ or RSS values from the user-space. The PDR values can be measured on any commodity hardware by measuring the received packets. It is possible that some environments such as iOS do not provide LQ or RSS information to the user-space. In such cases, *AMuSe* can rely on PDR measurements alone. As we show later, *AMuSe* with PDR measurements alone can provide reliable feedback.

B. Experiment Description

We now describe the types of experiments conducted to validate our hypotheses presented earlier in this section.

Different Bitrates: We fix the AP multicast transmission bitrate, denoted by TX_{AP}, to different values allowed by the card (6, 9, 12, 18, 24, 36, 48, 54 Mbps), each bitrate 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 bitrate 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 -70 dBm (low noise), we vary noise power in steps of 5 dBm up to -35 dBm (high noise).

Fig. 5 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, node (13,11) with $LQ = 20$ and $PDR = 0$ in Fig. 5(a) and 5(b) for a noise level at -70 dBm. 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) values 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) values. For instance, in Fig. 5(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 Testing

We turn to test our hypotheses based on the information collected from the experiments described in Section VI-B.

H1 - Correlation between PDR and LQ: Fig. 6(a)-6(e) demonstrate the correlation between the PDR of a node with respect to its LQ for different transmission bitrates without external noise, whereas, Fig. 6(f) shows the correlation between the PDR of a node with respect to its distance from the AP at a transmission rate of 48 Mbps. PDR values are close to 100% for almost all nodes for bitrates up to 24 Mbps (Fig. 6(a)-6(b)). Some degradation of PDR values is observed for bitrates of 36 Mbps (Fig. 6(c)) and even higher variance of PDR values are seen for 48 Mbps (Fig. 6(d)) and above.

Fig. 6(d) and Fig. 6(e) show 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. Fig. 6(f) illustrates very 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. 6(f) 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., [34], [35], [36] and [37].

H2 - Clustered nodes experience similar LQ and PDR: We measure the standard deviation (STD) of LQ and PDR without noise in each cluster radius of 3 and 6 meters on the grid, where each cluster contains an FB node and all its neighbors Histograms of the distribution of the LQ and PDR STD in different clusters are shown in Fig. 7(a)-7(d). We measure the same distributions in the presence of various noise levels with a cluster radius of 3 meters, and plot the results in Fig. 7(e) and Fig. 7(f). We expect the STD across clusters to be a good measurement of how similar the PDR and the LQ values are.

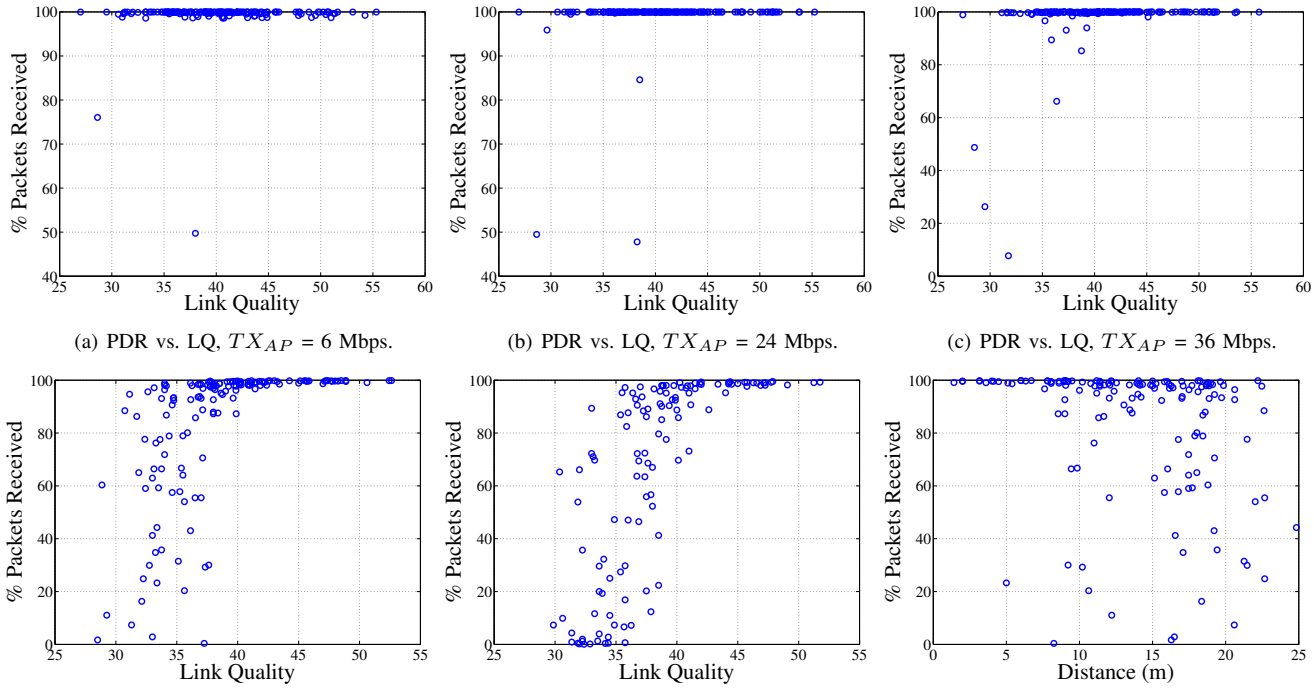
Fig. 7(a), Fig. 7(c), and Fig. 7(e) show that the LQ STD is very similar across all the bitrates regardless of the noise levels. This indicates that although adjacent nodes experience similar LQ (and similar RSS), the LQ metrics do not capture the effect of external interference and bitrate variation. By comparing Fig. 7(a) and Fig. 7(c), we see that a higher percentage of clusters report higher LQ STD for cluster size 6 m than with cluster size 3 m.

We now consider the distribution of the PDR STD values. Fig. 7(b) shows that with $TX_{AP} \leq 36$ Mbps, only very few clusters show significant deviations ($> 5\%$) in PDR. This is because most nodes have PDR above 99% when $TX_{AP} \leq 36$ Mbps as shown in Fig. 6. However, the variability of the PDR becomes evident at higher bitrates. By comparing Fig. 7(b) and Fig. 7(d), we observe that a higher percentage of clusters report higher PDR STD for cluster size 6 m as compared to cluster size 3 m. Further, we see in Fig. 7(d) that at higher bitrates, PDR STD is higher for a significant number of clusters.

As shown in Fig. 7(f), interference introduces noticeable deviations ($> 5\%$) in PDR across nearly two-thirds of the clusters. To understand this, we revisit the heatmaps in Fig. 5(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 are not able to decode the AP beacons (at a bitrate of 6 Mbps) disconnect from the AP, are not shown on the heatmap, and are not included in the variance calculations. The nodes which report a 0 PDR value are the ones that fail to receive any multicast packet. These nodes are shown in the heatmap red with a 0 value. At higher noise levels, many more nodes report PDR values of 0. This explains the high levels of PDR variance observed in Fig. 7(f).

The increase in LQ and PDR STD with the cluster size point to the inherent tradeoff in FB node selection process using both LQ and PDR as the quality metrics. The system should ideally operate in a mode where a large fraction of the nodes experience high PDR and the PDR STD is very low. Increasing the cluster size reduces the number of FB nodes, however, leads to increased STD of quality metrics in clusters, particularly the PDR STD at higher bitrates. The average number of FB nodes for different cluster sizes is shown in Fig. 8(a). The FB overhead of AMuSe is directly proportional to the number of FB nodes. Each FB node, periodically sends an FB message which is roughly 100 bytes long. The frequency of feedback messages is application-specific e.g., for multicast rate adaptation application, 1s could be sufficient [11]. This implies that 50 FB nodes will add an overhead of 40Kbps. In our case, 50 FB nodes correspond to a cluster radius of 3m from Fig. 8(a). The FB overhead is much smaller than the multicast throughput measured at the AP (order of Mbps even for bitrate of 6Mbps). The above observations serve as a good motivation to carefully set the parameters for the FB node selection algorithm.

Finally, we demonstrate that clustering is not redundant by comparing the proximity of channel quality values within and across clusters. Fig. 8(b) shows the CDF of the PDR differences between pairs of nodes inside and across clusters for bitrate of 54Mbps and no noise for a cluster radius of 3m. We chose bitrate of 54Mbps for ease of exposition. Roughly 60% of the node pairs have PDR differences less than 20% within a cluster while fewer than 50% of pairs have differences less than 20% across clusters. Similarly, Fig. 8(c) shows the CDF of the PDR differences between pairs of nodes inside and across clusters for bitrate of 12Mbps and external noise of -30dBm for a cluster radius of 3m. In this case also, the differences are similar. These results show that clustering is effective in grouping nodes with similar channel qualities.



(a) PDR vs. LQ, TX_{AP} = 6 Mbps. (b) PDR vs. LQ, TX_{AP} = 24 Mbps. (c) PDR vs. LQ, TX_{AP} = 36 Mbps.

(d) PDR vs. LQ, TX_{AP} = 48 Mbps. (e) PDR vs. LQ, TX_{AP} = 54 Mbps. (f) PDR vs. distance, TX_{AP} = 48 Mbps.

Fig. 6. Experimental results for testing hypothesis H1 and verifying the presence of abnormal nodes.

H3 - Clustered nodes suffer from similar interference: Fig. 5 shows that external noise has a largely local effect near the noise source. Moreoever, Fig. 7(f) shows that even with a small cluster size of 3 meters, the PDR STD can be high due to external interference. The above two observations validate the need for a well-distributed and non-sparse set of FB nodes to report the values of quality metrics in order to reflect the interference experienced by receivers.

Our experiments also show that increasing TX_{AP} has an impact on all nodes, and that beyond a certain bitrate, the PDR of many nodes drops below 90%, as shown in Fig. 6(d) and Fig. 6(f). Thus, it is critical to assign TX_{AP} appropriate values in order to improve the multicast service.

D. Abnormal Nodes

In general, we refer to a node with low PDR as *abnormal*. Specifically, in our experiments, a node is *abnormal* if its PDR is below the abnormal threshold $H = 90\%$. In contrast, a node is *normal* if its PDR is at least $H = 90\%$. In this section, we study the number of abnormal nodes as a function of the TX_{AP} and the link quality (LQ). Fig. 6(a)-6(d) show how PDR varies with LQ for each node in a single experiment run with TX_{AP} bitrates 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. 6(a)-6(c), PDR values are close to 100% for a large fraction of the nodes for bitrates up to 36 Mbps. However, Fig. 6(a) demonstrates that even in the extreme case of very low TX_{AP} without any interference some of the nodes (two in this case) are abnormal and suffer from low PDR.

The set of abnormal nodes remained small when we increase TX_{AP} to higher bitrates until 36 Mbps, as shown in Fig. 6(b) and Fig. 6(c). The number of abnormal nodes increases significantly once TX_{AP} reaches 48 Mbps. Surprisingly, the set of abnormal nodes is not the same in all experiments.

VII. Feedback Node Selection

The primary objective of this section is to study the performance of feedback node selection schemes. We compare AMuSe FB node selection system with other schemes and in the process, validate our *main hypothesis* from Section I. We consider the following schemes including the three flavors of AMuSe that select either the LQ, the PDR or a mix as the metric which is used by the AP for selecting FB nodes.

(i) AMuSe-LQ – AMuSe based on LQ.
(ii) AMuSe-PDR – AMuSe based on PDR.
(iii) AMuSe-Mix – AMuSe based on mix of LQ and PDR.
(iv) K-worst-LQ – K nodes with lowest LQ are FB nodes.
(v) K-worst-PDR – K nodes with lowest PDR are FB nodes.
(vi) K-random – K random nodes as FB nodes.

The AMuSe-Mix system relies on lexicographic ordering of PDR and LQ values for comparing channel quality. For nodes with PDR $> 98\%$, the ordering is based on LQ. For nodes with PDR $\leq 98\%$, the ordering is based on PDR. Thus, the channel quality is defined by the following tuple in lexicographic order: $(\min(PDR, 98), LQ)$ The motivation behind AMuSe-Mix lies in our observation that LQ is weakly correlated with PDR in Section VI. Very high PDR values ($> 98\%$) could result from random packet losses and small PDR variations above this value are unreliable indicators of difference in channel quality. Thus, we use AMuSe-Mix to study if LQ can be a better metric to distinguish nodes which have high PDR values.

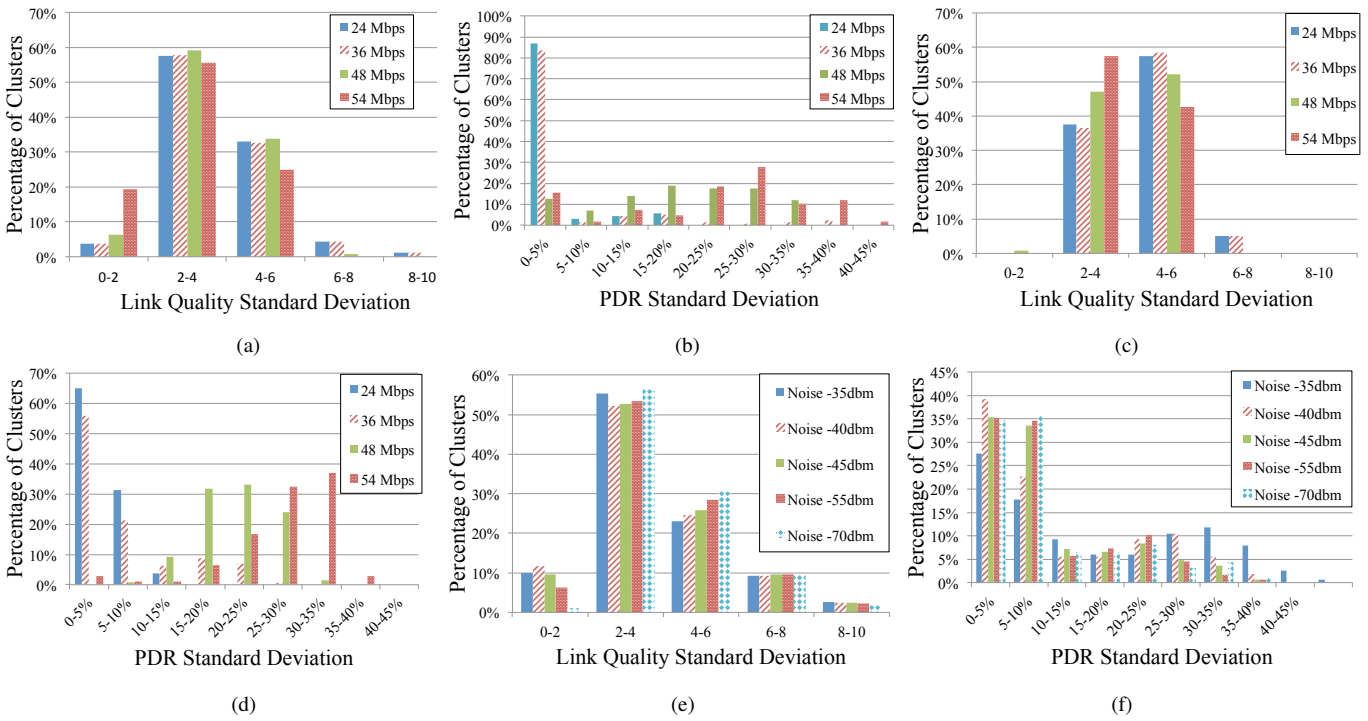


Fig. 7. Experimental results for testing hypotheses H2–H3: (a) LQ STD: varying TX_{AP} without noise, cluster size = $3m$, (b) PDR STD: varying TX_{AP} without noise, cluster size = $3m$, (c) LQ STD: varying TX_{AP} without noise, cluster size = $6m$, (d) PDR STD: varying TX_{AP} without noise, cluster size = $6m$, (e) LQ STD: varying noise, TX_{AP} = 12 Mbps, cluster size = $3m$, and (f) PDR STD: varying noise, TX_{AP} = 12 Mbps, cluster size = $3m$.

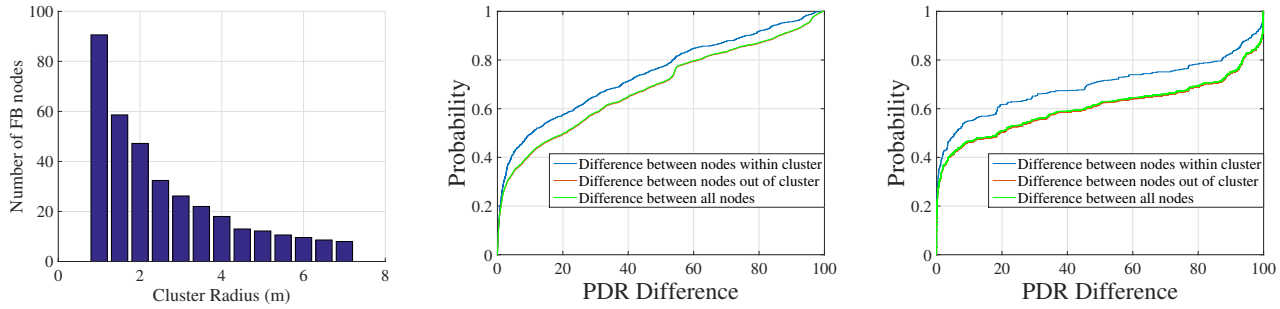


Fig. 8. The impact of clustering: (a) the number of FB nodes for different cluster sizes, (b) CDF of PDR differences of pairs of nodes within and across clusters for no external noise and bitrate of 54Mbps, and (c) CDF of PDR differences of pairs of nodes within and across clusters for external noise of -30dBm and bitrate of 12Mbps.

Moreover, we study the parameter choices for cluster radius (represented by the adjacency parameter, D). When we refer to cluster radius D as a parameter for the Random, K-worst-LQ, or K-worst-PDR schemes, we select as many FB nodes as AMuSe feedback schemes have (for a fair comparison).

We study the performance of different feedback nodes selection schemes under two network settings:

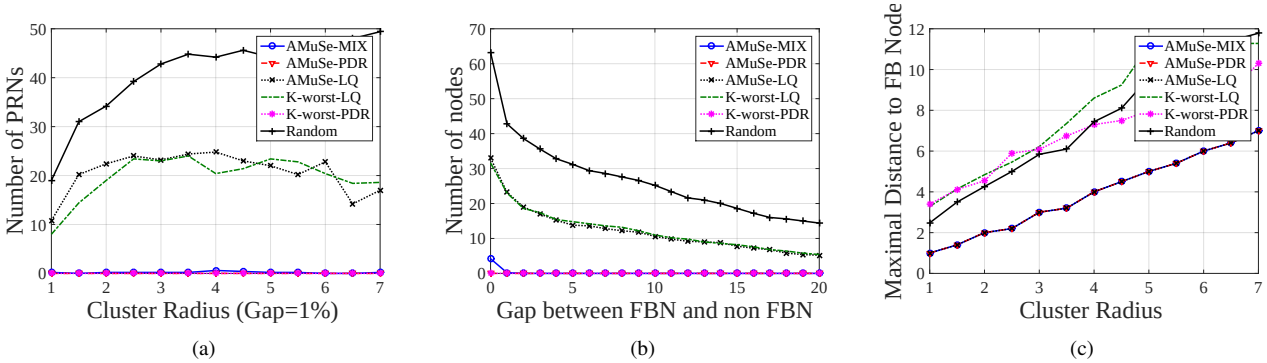
- **Static Settings:** The multicast bitrate and the external interference level are fixed.
- **Dynamic Settings:** In a dynamic environment of either (i) changing multicast bitrate, (ii) changing external interference, or (iii) emulated mobility.

For all our evaluations in both the static as well as the dynamic settings, we collected detailed packet traces at each node in the testbed for several bitrate and interference conditions. The number of nodes in the experiments was kept similar between 170 to 200 to avoid any performance mismatch. All the results for varying bitrate conditions were averaged over five runs of 10s at each bitrate. We ensured the appropriate setting of controlled interference by measuring the interference on a spectrum-analyzer on the testbed. During our experiments we observed sporadic spikes of uncontrolled interference. For mitigating their impact, we consider only time instants when there was no uncontrolled noise in our evaluations.

A. Static Settings

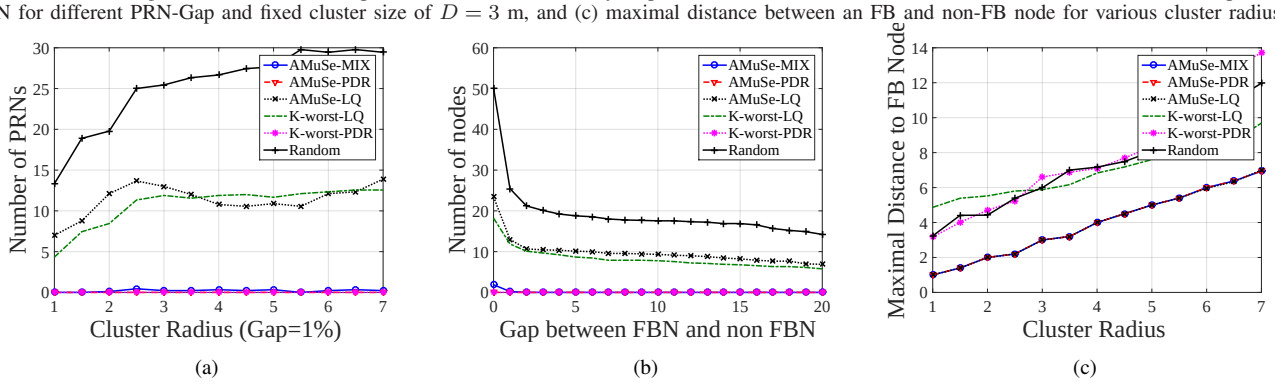
We first study the performance of different feedback schemes while the multicast bitrate and the generated external noise level are fixed. This setting allows us to evaluate the various schemes under normal network operation in stable conditions. We repeat our experiments with different bitrates and noise levels. We present our results for 3 different cases.

(i) **Fixed bitrate of** 36 **Mbps –** The optimal bitrate at which most of the nodes experience PDR close to 100 and only



(a) (b) (c)

Fig. 9. Static settings with bitrate of 48Mbps: (a) the number of Poorly Represented Nodes (PRN) vs. the cluster radius with fixed PRN-Gap of 1%, (b) PRN for different PRN-Gap and fixed cluster size of $D = 3$ m, and (c) maximal distance between an FB and non-FB node for various cluster radius.



(a) (b) (c)

Fig. 10. Static settings with external noise: (a) the number of Poorly Represented Nodes (PRN) vs. the cluster radius with fixed PRN-Gap of 1%, (b) PRN for different PRN-Gap and fixed cluster size of $D = 3$ m, and (c) maximal distance between an FB and non-FB node for various cluster radius.

a few nodes suffer from low PDRs, as shown in Fig. 6(c).

(ii) **Fixed bitrate of** 48 **Mbps –** Above the optimal bitrate many nodes experience low PDR, as shown in Fig. 6(f).

(iii) **External Noise –** The bitrate is set to 12 Mbps and the receivers suffer from different interference levels between -70dBm to -35dBm. The interference is concentrated on one corner of the grid as in Section VII-A.

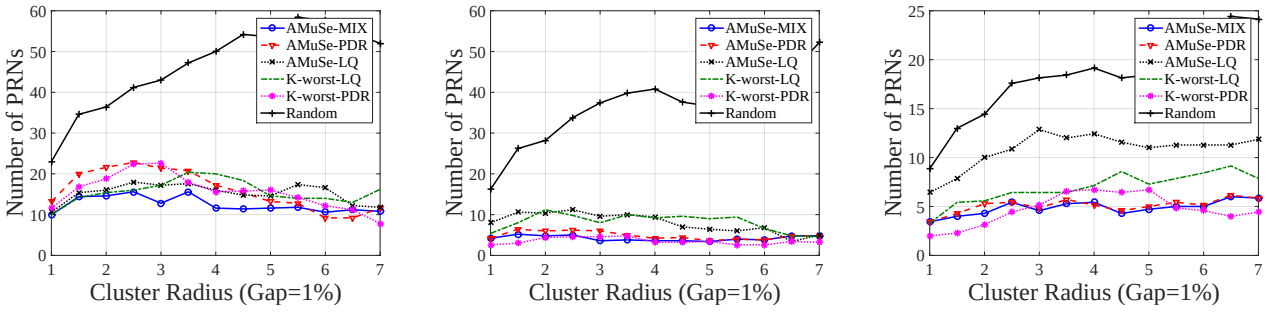
The results of our evaluation are presented in Figs. 9-10. Figs. 9(a) and 10(a) show the number of PRNs as the cluster radius D increases at bitrate 48Mbps without external noise and at bitrate of 12Mbps wit external noise respectively. We only show the nodes with minimum PRN-Gap of 1% to avoid counting non-FB nodes with PDRs lower than their associated FB nodes by a small margin as PRN. Both AMuSe-Mix and AMuSe-PDR yield close to 0 PRNs since both schemes select nodes with lowest PDR in each cluster. K-worst-PDR also yields 0 PRNs, since it selects nodes with overall lowest PDR values. The link quality based schemes AMuSe-LQ and K-worst-LQ have similar performance which could be explained due to the weak correlation between LQ and PDR. As expected, the Random feedback selection scheme performs the worst and as the number of feedback nodes decreases (increase in cluster size), the number of PRNs increases due to fewer feedback nodes. We omit the results at lower bitrates since they are qualitatively similar but yield fewer overall PRNs since the vast majority of the nodes experience PDR above 99%. The Random scheme yields much higher number of PRNs that increases with the cluster radius.

Figs. 9(b) and 10(b) present the number of PRNs at different values of PRN-Gap at bitrate 48Mbps without external noise and at bitrate of 12Mbps wit external noise respectively. The Random, K-worst-LQ, and AMuSe-LQ schemes result in a considerable number of PRNs. This number is high even for a PRN-Gap of 20% (e.g., Fig. 9(b) and 10(b) show that the K-worst-LQ and AMuSe-LQ schemes have between 5 to 10 PRNs with PRN-Gap of 20%). This means that the PDR value of each one of these nodes is at least 20% lower than its representative FB node. The situation is even worse for the Random scheme. We again omit the results at lower bitrates due to very low number of PRNs.

Finally, Figs. 9(c) and 10(c) show the maximum distance between an FB and non-FB node as D increases at bitrate 48Mbps without external noise and at bitrate of 12Mbps wit external noise respectively. As expected, for AMuSe schemes, this distance scales linearly with D. The maximum distance between an FB and non-FB node is significantly higher for the Random scheme and it is about twice for the K-worst-LQ and K-worst-PDR schemes. This indicates that FB nodes might be concentrated in areas of high losses. Thus, even though K-worst-PDR scheme leads to low number of PRNs, it does not obtain a well-distributed set of FB nodes. The distribution of FB nodes could be especially important in case of rapid network changes.

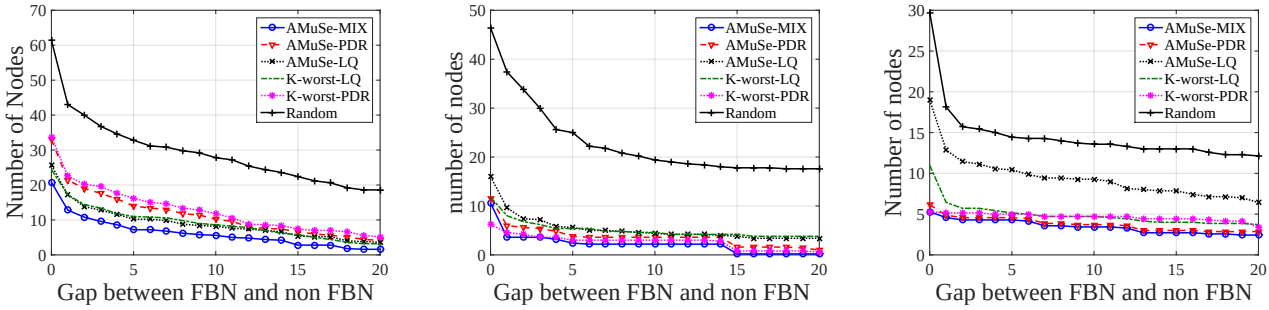
B. Dynamic Settings

Next, we emulate a dynamic environment of either: (i) changing AP bitrate, (ii) changing external interference, (iii) emulating node mobility. The methodology of the dynamic evaluations of (i) and (ii) relies on selecting a feedback set at one bitrate or external interference value and studying the performance of that set at a different value of bitrate or interference. Since the ORBIT environment is relatively static,



(a) Switching from 36 to 48 Mbps (b) Switching from 48 to 54 Mbps (c) Increasing noise by 10 dB

Fig. 11. Dynamic Settings: The number of Poorly Represented Nodes (PRN) vs. the cluster radius with fixed PRN-Gap of 1%.



(a) Switching from 36 to 48 Mbps (b) Switching from 48 to 54 Mbps (c) Increasing noise by 10 dB

Fig. 12. Dynamic Settings: The number of Poorly Represented Nodes (PRN) for different PRN-Gap and fixed cluster size of $D = 3$ m.

we emulate mobility by exchanging positions of nodes but keeping their channel quality values fixed. The FB nodes are selected at a particular setting and a fixed percentage of non-FB nodes exchange locations with each other within a certain radius. The PRNs are then evaluated with the same FB nodes and clustering as the initial conditions. The dynamic setting helps to evaluate the performance of the considered schemes under changes in the network.

Obviously, under such dynamic changes, the feedback node selection process may choose a new set of FB nodes. However, this process may require noticeable convergence time (depending on several parameters, such as τ_{AP} and τ_{FB}) of up to a few seconds. During this time the system may not receive accurate reports about the service quality. Thus, it is essential that the selected FB nodes continue to provide accurate FB reports in the event of such changes. For instance, during any interference episode, the AP should receive the accurate feedback information without delays to take appropriate interference mitigation actions, such as adding more FEC, reducing bitrate, etc. Similarly, if the AP increases the multicast bitrate using a rate adaptation algorithm, the FB nodes should provide accurate state information about the change to the AP. For the dynamic setting we consider the following cases: (a) Switching from bitrate of 36 Mbps to 48 Mbps, (b) Switching from bitrate of 48 Mbps to 54 Mbps, (c) Increasing the noise level by 10 dB, and (d) Emulated mobility.

Fig. 11 presents the number of PRNs vs. the cluster radius (D) for the three cases where the PRN-Gap is 1%. Fig. 11(a) shows the number of PRNs when switching the bitrate from 36 to 48 Mbps. In this case, the AMuSe-LQ and K-worst-LQ have comparable performance to the static case with bitrate of 48 Mbps. This is an expected result since LQ is a measure of the received signal strength and is not affected from changing the bitrate. However, AMuSe-PDR performs significantly worse than the static case. To understand this trend, recall that at bitrate of 36 Mbps most of receivers experience PDR close to 100%, as shown in Fig. 6(c). Therefore, when the cluster size is small and large number of receivers are selected as FB nodes, most of the FB nodes have PDR above 99%. With such high PDR, a selected FB node may not be affected by increasing the bitrate. Observe that the number of PRNs decreases by increasing the cluster size. This is not surprising since now most of the selected FB nodes have PDR below 98%, which indicates that they experience only moderate channel quality and therefore they are more susceptible to a bitrate increase. A similar explanation holds true for the K-worst-PDR scheme. AMuSe-Mix outperforms the other schemes since it considers both the PDR and the LQ of the receivers and uses the LQ values when the PDR is very high. Like the static setting, the Random scheme suffers from very high number of PRNs.

Fig. 11(b) shows the number of PRNs for bitrate increases from 48 to 54Mbps. In this case AMuSe-Mix , AMuSe-PDR , and K-worst-PDR outperform the LQ based solutions. By revisiting Fig. 6(f), we see that many receivers suffer from low PDR due to a weak channel condition at a bitrate of 48 Mbps. Since these nodes are selected as FB nodes, they provide good lower bound reports of the service quality observed by the nodes in their clusters. We notice a similar situation in Fig. 11(c) when increasing the noise level by 10 dB.

The distribution of PRNs vs the PRN-Gap is shown in Fig. 12 for a cluster radius $D = 3$ m. The figure supports our observations from Fig. 11 and demonstrates that AMuSe-Mix outperforms the other alternatives in all cases. Since the feedback node set is not changed when increasing the bitrate

or noise level, the maximum distance between an FB and non-FB node remains the same as shown in Figs. 9(c) or 10(c).

The results for emulated node mobility are shown in Fig. 13. Fig. 13(a) shows the number of PRNs vs. the percentage of moved nodes within a radius of 2m at a fixed bitrate of 36Mbps. Similar results at bitrate of 48Mbps are in Fig. 13(b) and with external noise in Fig. 13(c). The Random scheme yields the largest number of PRNs and is not affected by increasing number of moved nodes. The AMuSe-Mix, AMuSe-PDR, and K-worst-PDR schemes perform quite similarly and the PRNs for all of them increase with increase in the number of moved nodes. The LQ based schemes AMuSe-LQ and K-worst-LQ perform worse than the PDR based schemes.

We also 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. However, we observed only insignificant increases in the number of PRNs for the AMuSe schemes.

Our experiments on the ORBIT testbed with approximately 200 nodes validate the practicality of AMuSe-Mix as an excellent scheme for reporting the provided quality of an ongoing WiFi multicast services for both static and dynamic settings. The K-worst-PDR scheme also peforms quite well but does not yield a well-distributed set of FB nodes. Our evaluation shows that a relative small number of FB nodes is sufficient to provide accurate reports. Yet, the number of required FB nodes will also depend on the application.

VIII. Conclusion and Future Work

In this paper, we present the design and large-scale experimental evaluation of the AMuSe system for providing scalable and efficient quality monitoring of WiFi multicast services to a large group of users. AMuSe only needs access to the channel quality measurements such as RSSI and Packet Delivery Ratio on WiFi devices and can be implemented as an application layer protocol on existing devices.

IX. Acknowledgments

We thank Ivan Seskar for his critical support. This work was supported by (while one of the authors was serving at) the NSF, 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, the Generalitat de Catalunya under grant 2009-SGR-940 and the People Programme (Marie Curie Actions) of the European Unions Seventh Framework Programme (FP7/20072013) under REA grant agreement no. [PIIF-GA-2013-629740].11. Any opinion, findings and conclusions or recommendations expressed here do not necessarily reflect the views of the NSF.

References

[1] Y. Bejerano, J. Ferragut, K. Guo, V. Gupta, C. Gutterman, T. Nandagopal, and G. Zussman, "Scalable wifi multicast services for very large groups," in *Proc. IEEE ICNP'13*, 2013.

[2] ——, "Experimental evaluation of a scalable WiFi multicast scheme in the ORBIT testbed," in *Proc. GENI Research and Educational Experiment Workshop (GREE14) (invited)*, 2014.

[3] "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.

[4] Y. Tanigawa, K. Yasukawa, and K. Yamaoka, "Transparent unicast translation to improve quality of multicast over wireless LAN," in *Proc. IEEE CCNC'10*, 2010.

[5] "Cisco, white-paper, Cisco connected stadium Wi-Fi solution," 2011. [Online]. Available: http://www.cisco.com/web/strategy/docs/sports/c78-675064_svcs.pdf

[6] "Yinzcam," http://www.yinzcam.com/.

[7] 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.

[8] N. Hajlaoui and I. Jabri, "On the performance of IEEE 802.11n protocol," in *Proc. ACM WiNTECH'12*, 2012.

[9] C. Perkins and M. McBride, "Multicast WiFi problem statement," Working Draft, IETF Internet-Draft, 2015, https://tools.ietf.org/html/draft-mcbride-mboned-wifi-mcast-problem-statement-00.

[10] 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.

[11] V. Gupta, C. Gutterman, Y. Bejerano, and G. Zussman, "Experimental evaluation of large scale wifi multicast rate control," in *Proc. IEEE INFOCOM'16*, 2016.

[12] 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.

[13] "ORBIT testbed," http://orbit-lab.org/.

[14] 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.

[15] N. Choi, Y. Seok, T. Kwon, and Y. Choi, "Leader-based multicast service in IEEE 802.11v networks," in *Proc. IEEE CCNC'10*, 2010.

[16] J. Vella and S. Zammit, "A survey of multicasting over wireless access networks," *Communications Surveys Tutorials, IEEE*, vol. 15, no. 2, pp. 718–753, Second 2013.

[17] 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.

[18] J. K. Kuri and S. Kumar, "Reliable multicast in multi-access wireless LANs," *ACM/Kluwer Wirel. Netw.*, vol. 7, pp. 359–369, 2001.

[19] 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.

[20] 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.

[21] 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.

[22] 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.

[23] 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.

[24] S. K. S. Gupta, V. Shankar, and S. Lalwani, "Reliable multicast MAC protocol for wireless LANs," in *Proc. IEEE ICC'03*, 2003.

[25] X. Wang, L. Wang, and D. Wang, Y.and Gu, "Reliable multicast mechanism in WLAN with extended implicit MAC acknowledgment," in *Proc. IEEE VTC'08*, 2008.

[26] V. Srinivas and L. Ruan, "An efficient reliable multicast protocol for 802.11-based wireless LANs," in *Proc. IEEE WoWMoM'09*, 2009.

[27] S. Sen, N. K. Madabhushi, and S. Banerjee, "Scalable WiFi media delivery through adaptive broadcasts," in *Proc. USENIX NSDI'10*, 2010.

[28] Z. Feng, G. Wen, C. Yin, and H. Liu, "Video stream groupcast optimization in WLAN," in *Proc. IEEE ITA'10*, 2010.

[29] J. Villalon, P. Cuenca, L. Orozco-Barbosa, Y. Seok, and T. Turletti, "Cross-layer architecture for adaptive video multicast streaming over

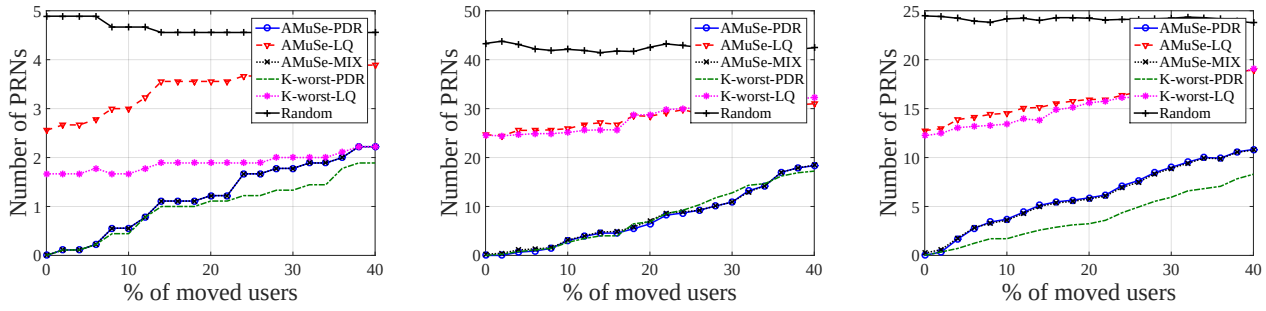


Fig. 13. The number of Poorly Represented Nodes (PRNs) vs. percentage of moved nodes for (a) fixed bitrate of 36Mbps, (b) fixed bitrate of 48Mbps, and (c) bitrate of 12Mbps and noise of 5dBm.

multirate wireless LANs," *IEEE J. Sel. Areas Commun.*, vol. 25, no. 4, pp. 699–711, 2007.

[30] 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.

[31] 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.

[32] 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.

[33] K. Chintalapudi, A. Iyer, and V. Padmanabhan, "Indoor localization without the pain," in *Proc. ACM MOBICOM'10*, 2010.

[34] H. Rahul, F. Edalat, D. Katabi, and C. G. Sodini, "Frequency-aware rate adaptation and MAC protocols," in *Proc. ACM MOBICOM'09*, 2009.

[35] 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.

[36] 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.

[37] H. Gudmundsdottir, E. I. Ásgeirsson, M. H. L. Bodlaender, J. T. Foley, M. M. Halldórsson, and Y. Vigfusson, "Wireless scheduling algorithms in complex environments," in *Proc. ACM MSWiM'14*, 2014.

Varun Gupta completed his undergraduate studies in electrical engineering from the Indian Institute of Technology, Delhi in 2011 and received the M.S. degree in Electrical Engineering from Columbia University, New York in 2012. He is currently working towards his Ph.D. at Columbia University. His research interests include network performance evaluation, stochastic modeling, wireless networks, and datacenter networks.

Yigal Bejerano is currently a member of the technical staff (MTS) at Bell Laboratories, Nokia. He received his B.Sc. in Computer Engineering in 1991 (summa cum laude), his M.Sc. in Computer Science in 1995, and his Ph.D. in Electrical Engineering in 2000, from the Technion - Israel Institute of Technology, Haifa, Israel. His research interests are management aspects of high-speed and wireless networks. He has published over 60 peer-reviewed research papers at the leading conferences and journals of the networking community. Dr. Bejerano is on the technical program committee (TPC) of numerous conferences and he also serves as an Associate Editor of the IEEE/ACM Transactions on Networking.

Craig Gutterman received the B.Sc. degree in electrical engineering from Rutgers University, New Jersey in 2012 and received the M.S degree in Electrical Engineering from Columbia University, New York in 2014. He is currently working towards his Ph.D. at Columbia University. He received the NSF GRFP and the From Data to Solutions NSF IGERT fellowships. His research interests are in the areas of wireless communications, networks, sensors, with applications to data analysis and exploration.

Jaime Ferragut holds an M.Sc. and a Ph.D. in Electrical Engineering from the Universitat Politcnica de Catalunya BarcelonaTech. From 2009 to 2014 he worked as a Research Assistant and Ph.D. Fellow at CTTC (Barcelona, Spain). Since August 2014 he works as an Advanced Product Development Engineer at ARM (Cambridge, UK). His research focuses on data plane optimisation for next-generation mobile communications (C-RAN, 5G).

Katherine Guo is with the IP Platforms Research Program at Bell Laboratories, Nokia. She has extensive research and product experience in Multimedia Streaming, Internet Content Distribution, Multicasting, Cellular Wireless Systems, IP Multimedia Subsystems (IMS), Distributed Systems, Cloud Computing, Quality of Service support for real time applications such as VoIP, Video Streaming and Distributed Gaming. She has served as the architect for Lucent's IPWorX/Imminet streaming cache and video content distribution product line. She received her Ph.D. degree in computer science from Cornell University in Ithaca, New York in 1998.

Thyaga Nandagopal serves in the Directorate of Computer and Information Science and Engineering (CISE) of the National Science Foundation. He manages wireless networking and mobile computing research within the Networking Technologies and Systems (NeTS) program at NSF. He has been with the Foundation since February 2012. Dr. Nandagopal received his Ph.D. in Electrical Engineering in 2002 from the University of Illinois at Urbana-Champaign. He was at Bell Labs from 2002 to 2012. His research interests have spanned several areas over these years: wireless ad hoc/mesh networks, RFID/sensor networks, internet routing architectures and protocols, and cloud computing.

Gil Zussman received the Ph.D. degree in electrical engineering from the Technion in 2004 and was a postdoctoral associate at MIT in 2004–7. He is currently an Associate Professor of Electrical Engineering at Columbia University. He is a corecipient of 5 paper awards including the ACM SIGMETRICS'06 Best Paper Award and the 2011 IEEE Communications Society Award for Outstanding Paper on New Communication Topics. He received the Fulbright Fellowship, the DTRA Young Investigator Award, and the NSF CAREER Award, and was a member of a team that won first place in the 2009 Vodafone Foundation Wireless Innovation Project competition.

APPENDIX

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

Proof of Proposition 1: We prove the general proposition of $|F| \leq 5 \cdot |OPT|$, which is based on Lemma 3.1 in [12]. The special case of $|F| \leq 3 \cdot |OPT|$, where the channel quality is a monotonic decreasing function with the distance from the AP, can be proved by using similar arguments and Lemma 3.3 in [12].

Consider a point x in the plane and let Z be an independent set of points in the circle with radius r around point x. i.e, the distance between any two points in Z is more than r. Then according to Lemma 3.1 in [12], $|Z| \leq 5$.

To prove that AMuSe guarantees approximation ratio of 5, we just need to show that for any given multicast group there is a mapping from F to OPT such that at most 5 nodes in F are mapped to the same node in OPT. To this end, we map every FB node $v \in F$ to its nearest node $u \in OPT$, which may be node v itself. Recall that both OPT and F are dominating independent sets such that each node has an adjacent FB node with distance at most D and the minimal distance between any pair of FB nodes is at least D. From this it is implied that any FB node v is either in OPT or it is D-adjacent to at least one node in OPT.

Now, consider an FB node $u \in OPT$ and let $W \subseteq F$ be the set of FB nodes selected by our scheme that are D-adjacent to u. Since F is an independent set it holds that W is also an independent set, i.e., the minimal distance between any pair of FB nodes $x, y \in W$ is $d_{x,y} > D$. Observe that all nodes in W are included in a disk with radius D centered at node u. Thus, according to Lemma 3.1 in [12], it follows that $|W| \leq 5$. This leads to the result that each node in OPT is associated with at most 5 nodes in F. ■