BoLTE: Efficient Network-wide LTE Broadcasting

Rajarajan Sivaraj[†], Mustafa Arslan[*], Karthikeyan Sundaresan[*], Sampath Rangarajan[*], and Prasant Mohapatra[†]

[†]University of California, Davis, CA, [*]NEC Laboratories, Princeton, NJ

{rsivaraj, pmohapatra}@ucdavis.edu, {marslan, karthiks, sampath}@nec-labs.com

***Abstract*—Evolved-Multimedia Broadcast Multicast Services (eMBMS) is a set of features in LTE networks to deliver bandwidth-intensive multimedia content on a point-to-multipoint basis to subscribers. The notion of a *Single Frequency Network* (SFN) in eMBMS allows base stations to synchronize and transmit signals in a coordinated fashion across the same frequency-time radio resources using a common modulation rate. While SFN boosts the channel quality of users via transmit diversity gain, the use of a common rate across base stations results in reduced utilization for those that can individually support much higher data rates for their users, even without the notion of an SFN. Excluding such base stations from the SFN helps them utilize their resources better by not being constrained by the common rate, but creates additional inter-cell interference from their independent transmissions. Striking a balance between SFN cooperation and resource utilization is crucial for efficiently delivering broadcast content as well as other unicast flows. We design *BoLTE*, which carefully addresses this tradeoff and evaluate it using a prototype implementation over an SFN testbed, realized over a cloud-based radio access network system, as well as large-scale NS3 simulations. We show that *BoLTE* improves overall system throughput by around 40%.**

I. Introduction

3GPP LTE is the latest wireless broadband standard that offers higher data rates than its predecessors. With increasing popularity of video streaming applications like NetFlix, YouTube, etc. among billions of smartphone users, wireless video broadcast has become one of the most-demanding applications made viable with LTE using its IP-based evolved packet core. Evolved Multimedia Broadcast Multicast Services (eMBMS) are a set of technical specifications that allow LTE operators to deliver broadcast video to multiple subscribers over a wide geographical area by effectively utilizing the available bandwidth. Some of the envisioned use cases of eMBMS include live TV streaming, advertisements for video on-demand, events coverage/replays, emergency broadcast for public safety [3]. With growing consumer interest in broadcast video, estimated to form 70% of online traffic by 2018 [9], leading operators around the world (such as AT&T, Verizon, China Telecom, etc.) are investing in eMBMS and looking for ways to monetize it. However, the only commercial eMBMS deployment is carried by operator KT in South Korea, while most of the others are only in trial phase, since they are not yet armed with sophisticated tools to offer high quality video broadcast service. Our paper helps in understanding the trade-offs to design an efficient eMBMS system.

A. Definitions: eMBMS, SFN and bottlenecks

eMBMS: eMBMS enables point-to-multipoint broadcasting [1] of data in LTE, unlike the traditional point-to-point unicast delivery. In unicast, an LTE base station (called eNB) allocates a different set of transmission resources to serve each user equipment (UE) for its subscribed traffic, encoded with a distinct modulation (MCS) rate, supported by the UE's SINR. Even if multiple UEs subscribe to the same content, they compete for the finite spectrum resources available at the eNB (so as to be independently scheduled on distinct sets of resources). This results in ineffective spectrum utilization, critical to system performance, especially when the subscribed content is bandwidth-intensive (such as video). The spectrum resources are rapidly exhausted, causing starvation to a significant number of UEs. On the other hand, eMBMS allows an eNB to allocate *a common set of resources, encoded with a common rate*, for all UEs subscribing to the same video session. This point-to-multipoint nature of transmission is more scalable for delivering high-bandwidth services to large subscribers.

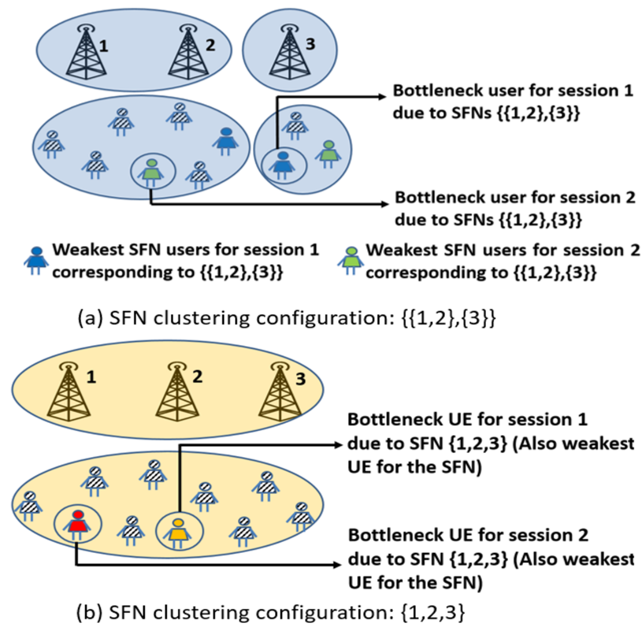


Fig. 1. Instantaneous weakest UEs of SFNs and session subscribers.

SFN: eMBMS offers a feature called *single frequency network* (SFN), where a cluster of eNBs are synchronized (at an OFDM symbol level) to transmit the same data over a common set of frequency-time radio resources encoded by a common rate to network-wide subscribers of a given content. With tight coordination across eNBs and the synchronous transmission of the same signal from multiple eNBs, the otherwise unwanted interference among them becomes constructive and the identical signals combine at the UEs, boosting their SINR. This technique is called transmit diversity and is especially useful

[1]Note that the terms broadcast and multicast are used inter-changeably throughout the paper.

978-1-5090-6501-1/17/$31.00

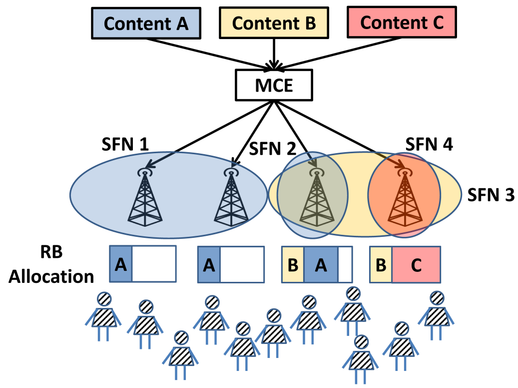


Fig. 2. MCE coordinates SFN transmission for different sessions.

in dense deployments, where inter-cell interference adversely impacts the performance of UEs, especially at the cell-edges.

Session-specific SFN common rate: The common rate selected by an SFN to serve a broadcast session on a point-to-multipoint basis is the minimum rate supported by any subscriber attached to any of the constituent eNBs in the SFN. This subscriber (supporting the minimum rate) is the instantaneous *weakest session subscriber served by the SFN*, based on the instantaneous combination of eNBs forming the SFN. Note that the weakest session subscriber in an SFN varies with different combination of eNBs constituting the SFN. This is due to change in diversity gains and inter-cell interference, with changing SFN clustering combinations. Selection of this common minimum rate guarantees delivery to all UE subscribers served by the SFN.

Session throughput: The throughput of a session is limited by the minimum throughput reported on any of its subscribers across the network. Since a broadcast session can be served by more than one SFN cluster, the subscriber with the minimum throughput among the weakest subscribers from each SFN cluster serving the session becomes the *net weakest session subscriber*. Its throughput is the session throughput.

B. Challenges in video broadcasting

Our paper talks about designing an efficient and scalable LTE broadcast system, delivering multiple video sessions and unicast traffic across a large subscriber base. Here, we identify some pertinent challenges and trade-offs:

(i) **Transmit Diversity vs Resource Multiplexing tradeoff**: Grouping more eNBs in an SFN increases its common rate by boosting the SINR of all UEs due to diversity gain. However, some eNBs could individually yield high SINRs for their UEs even without getting coupled in an SFN, thereby not benefiting much from the SFN. In fact, since an SFN binds its constituent eNBs to the common minimum rate on common resources (based on the weakest session subscriber), such eNBs will experience a degraded performance, if the common SFN rate is less than what they could individually yield to their UEs. In brief, even as grouping more eNBs in an SFN increases its common minimum rate, it also risks reducing the rates allocated by individual eNBs for their flows, which limits the ability of individual eNBs to effectively multiplex multiple broadcast/unicast traffic flows on their resources.

(ii) **Serving multiple flows**: A broadcast session can be served by more than one SFN cluster, however there should not be any overlapping eNBs between SFN clusters transmitting on the same radio resources (since each SFN can use different rates/resources than the other). Note that (i) the rate and resource allocation per SFN cluster is limited by the weakest session subscriber served by the SFN, and (ii) the performance of any session is limited by the net weakest subscriber served by any SFN. However, the net weakest session subscriber and the weakest UEs of the session-specific SFN clusters are *instantaneous*, meaning that they change with varying combinations of eNBs forming the SFN clusters. This is shown in Fig. 1. With 3 eNBs $\{1, 2, 3\}$, we consider 2 SFN combinations: $\{1, 2, 3\}$ and $\{\{1, 2\}, \{3\}\}$ serving 2 sessions. The weakest session subscribers in SFNs and the net weakest subscribers of the sessions vary between the 2 clustering combinations. Moreover, the combinations of SFN clusters vary across sessions. Hence, the rate/resource allocation pertaining to any session across its SFN clusters affects resource scheduling for other sessions and unicast traffic flows, served by the same eNBs.

(iii) **Synchronizing base stations**: The constituent eNBs in an SFN need to be tightly-synchronized at an OFDM symbol level to leverage common rate/resource allocation to transmit data. This requires a very low-latency/high-capacity control plane and a centralized controller, called Multicast Coordination Entity (MCE), to enable coordination between eNBs. Moreover, this tight synchronization is essential in switching between different SFN combinations, serving different sessions, across the sub-frames of a single OFDM frame. MCE is shown in Fig. 2, where there are 4 eNBs serving 3 sessions. The MCE is responsible for forming SFN clusters, synchronizing eNBs and allocating resources from eNBs to serve the session.

C. Our Contributions

Our paper addresses the above challenges for video broadcast in LTE systems. We propose an SFN design framework *BoLTE* to maximize performance of an LTE system, serving multiple broadcast sessions and unicast flows. Our contributions towards the design of *BoLTE* are multi-fold:

1. We determine the optimal set of SFN clusters for each session by managing the diversity - multiplexing tradeoff. Our solution can scale across any number of session subscribers.

2. We multiplex different traffic flows on the spectrum resources deployed across coordinated sets of eNBs. Our solution can scale across any number of flows.

3. We use a Cloud-based Radio Access Network (C-RAN) to realize an SFN. A centralized baseband controller performs rate/resource allocation across synchronized sets of Remote Radio Heads (RRH). We build a low-latency control plane to enable tight synchronization between the RRHs.

4. We evaluate a prototype of *BoLTE* using the C-RAN testbed and our design algorithms for SFN clustering and resource allocation using large-scale NS3 simulations for broadcast/ unicast performance, and show 40% improvements.

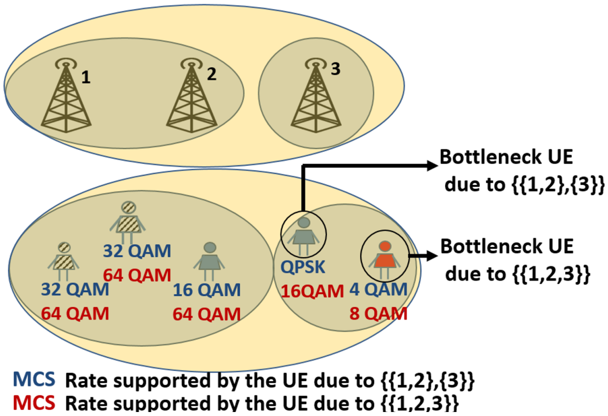


Fig. 3. Varying weakest UEs of SFNs and weakest session subscriber with different SFN combinations.

II. Model and Challenges

A. System Model

We consider an LTE network with a set N of eNBs with identical configurations such as central frequency, bandwidth and omni-directional antennae. Each eNB has B radio resources per LTE frame. There is a set K of broadcast sessions as part of the eMBMS service. Each session $k \in K$ can be bursty, imposing a per-frame finite buffer bit-rate, denoted by R_k. Each UE subscribes to either a broadcast session in K or a unicast flow. The set of UEs subscribing to a session k forms the eMBMS group for that session. The set M denotes the UEs with unicast flows. UEs are uniformly-distributed and associate to that eNB yielding the strongest signal. The UEs are assumed to be stationary within the granularity of a frame.

B. Trade-off

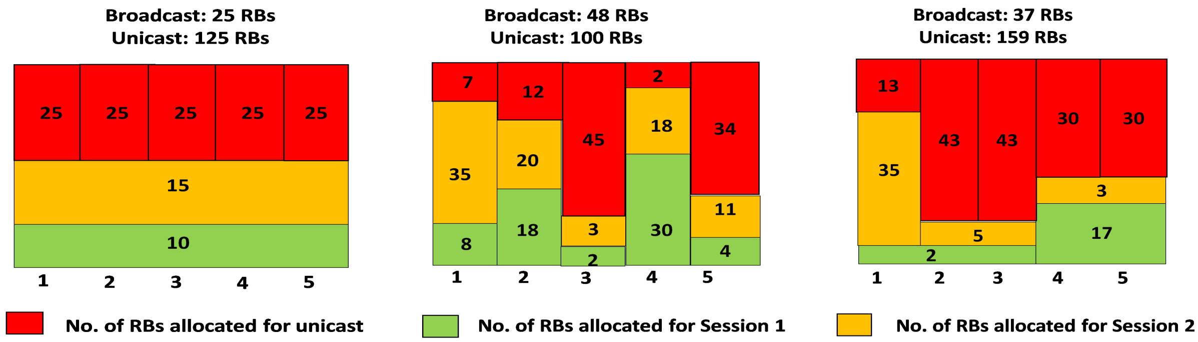
We discuss two conditions concerning the diversity-multiplexing trade-off. Let $u_k(C)$ denote the weakest UE for any SFN $C \subset N$ serving any session k.

1. The largest-possible SFN serving k is the one synchronizing the entire set N of eNBs in the network. Then, $u_k(N)$ is the weakest UE not only for the SFN N but also for the session k. Let $F_{k,C}$ be the common rate yielded by SFN C for session k. For any non-overlapping set $\mathbb{C}_k$ of smaller SFN clusters serving k, the rate corresponding to the instantaneous weakest session subscriber is given by $\min_{C \in \mathbb{C}_k} \{F_{k,C}\}$. Now, due to transmit diversity [7], $F_{k,N} > \min_{C \in \mathbb{C}_k} \{F_{k,C}\}$, since $C \subset N$. *That is, the instantaneous weakest subscriber has the highest rate with the single largest SFN N than that with any other SFN clustering combination due to maximum diversity gain.*

2. On the other hand, it is possible that $\exists C' \subset N$ such that $F_{k,C'} > F_{k,N}$. *That is, with varying weakest UEs across different SFN clusters with different clustering combinations, a smaller SFN C' could yield a higher common rate than the largest one N.* In such cases, $u_k(C') \neq u_k(N)$, since $u_k(N)$ experiences maximum rate (due to maximum diversity gain) only from the single-largest SFN.

This is shown in Fig. 3 using two SFN clustering combinations: (i) $\{1,2,3\}$ and (ii) $\{\{1,2\},\{3\}\}$. The modulation rates of UEs served by SFN $\{1,2,3\}$ and $\{\{1,2\},\{3\}\}$ are shown in red and blue respectively. The weakest UEs of the SFN clusters, serving the session, as a result of combinations $\{1,2,3\}$ and $\{\{1,2\},\{3\}\}$, are shaded in red and blue respectively. The net weakest session subscribers, as a result of combinations $\{1,2,3\}$ and $\{\{1,2\},\{3\}\}$, are circled. Note that, the net weakest session subscriber as a result of $\{1,2,3\}$ reports a higher rate of 8 QAM than the one due to $\{\{1,2\},\{3\}\}$, reporting QPSK (depicting the first condition) - benefiting from diversity gain. However, the weakest UE of the SFN $\{1,2\}$ reports a higher rate of 16 QAM than the one from the SFN $\{1,2,3\}$ reporting 8 QAM (depicting the second condition) - resulting in multiplexing gain. Also, note that the net weakest subscribers due to $\{1,2,3\}$ and $\{\{1,2\},\{3\}\}$ are different. Note that, with single-cell broadcast, $\min_{C \in \mathbb{C}_k} \{F_{k,C}\} >= \min_{n \in N} \{F_{k,\{n\}}\}$, since the diversity gain is the minimum without the notion of SFN. In other words, the net weakest session subscriber experiences the minimum rate in the case of single-cell broadcast, while similarly, it is possible that $\exists n \in N$ such that $F_{k,\{n\}} >= \min_{C \in \mathbb{C}_k} \{F_{k,C}\}$. Smaller SFN clusters meeting the second condition, though resulting in reduced diversity gain for the net weakest session subscriber, could yield better multiplexing gains as all eNBs are not constrained by the same rate/resource allocation, unlike the single large SFN. This increases the overall performance, especially with multiple broadcast/unicast flows.

Consider a system with 5 eNBs $\{1,2,3,4,5\}$ - each with 50 radio resources (called RBs), 2 broadcast sessions K_1 and K_2 with required bit rates R_1 and R_2 and unicast traffic flows. The single large SFN, shown in Fig. 4(a), $\{1,2,3,4,5\}$ yields the maximum diversity gain for the net weakest subscribers to both sessions. The SFN requires 10 and 15 resources to serve sessions K_1 and K_2 for rates R_1 and R_2, respectively. The remaining 25 resources on each of the 5 eNBs (150 resources, in total) are used to schedule unicast traffic flows. For single-cell broadcast in Fig. 4(b), it requires 30 and 35 resources for the net weakest session subscribers of K_1 and K_2 to serve R_1 and R_2 respectively. Since the diversity gain and hence, the rate per resource, is minimum for the net weakest session subscribers due to single cell broadcast, it takes more resources to serve the same amount of traffic. However, due to the second condition, there are some eNBs that require fewer than 10 and 15 resources to serve R_1 and R_2, respectively. But, the total number of resources for unicast flows is 100, less than in Fig. 4(a), since the loss in diversity gain is not compensated by resource multiplexing gain. The optimal SFN clustering solution is the one that ideally balances the diversity-multiplexing tradeoff (lying intermittently between the single-large SFN and single-cell broadcast), as illustrated in Fig. 4(c). The optimal SFN clustering solutions for the sessions K_1 and K_2 are $\{\{1,2,3\},\{4,5\}\}$ and $\{\{1\},\{2,3\},\{4,5\}\}$. While the net weakest session subscribers give up a bit on their diversity gain (their SFN clusters allocate more than the 10 and 15 resources seen in Fig. 4(a)), the other SFN clusters ($\{1,2,3\}$ for K_1 and $\{\{2,3\},\{4,5\}\}$ for K_2) yield higher common rates, and hence better multiplexing gains, than the single-large SFN clusters seen in Fig. 4(b). Balancing the diversity-multiplexing tradeoff, the total number of resources for unicast



(a) Single-large SFN (b) Single-cell bcast (c) SFN Clustering

Fig. 4. Illustrating the tradeoff between diversity and multiplexing.

flows is 159, greater than in Figs. 4(a,b).

III. Objective and Problem Formulation

The core challenge in the design of LTE broadcast systems is to determine the optimal session-specific SFN clusters by managing the diversity-multiplexing trade-off. We address this by adopting an optimization approach for system utility maximization. Our objective is to maximize the aggregate system utility of unicast and broadcast flows in each frame. The utility of a flow is based on proportional fairness (PF) [8], defined as the $\log$ of its throughput. A higher PF utility ensures serving more sessions with higher throughput. For a broadcast session, we define the utility as the $\log$ of its session throughput, defined as the achieved throughput of the net weakest session subscriber. The utility is defined on a per-frame basis, since the bursty nature of traffic affects the amount of data served per frame.

$$
\begin{aligned}
\text{Maximize} \quad & \sum_{k \in K} \alpha_k \log A_k + \sum_{m \in M} \sum_{n \in N} \alpha_m \log(Q_{m,n} Y_{m,n}) \\
\text{subject to} \quad & F_{k,C} X_{k,n} \geq A_k,\ \forall k \in K, C \in \mathbb{C}_k, n \in C \\
& F_{k,C} X_{k,n} \leq R_k,\ \forall k \in K, C \in \mathbb{C}_k, n \in C \\
& \sum_{n \in N} Q_{m,n} Y_{m,n} \leq R_m,\ \forall m \in M \\
& \sum_{k \in K} X_{k,n} \leq \theta \cdot B, \forall n \in N \\
& \sum_{k \in K} X_{k,n} + \sum_{m \in M} Y_{m,n} \leq B, \forall n \in N \\
& X_{k,n} \in \{0, 1, \dots \theta \cdot B\}, Y_{m,n} \in \{0, 1, \dots B\} \\
& \sum_{n \in N} X_{k,n} \geq 1, \sum_{n \in N} Y_{m,n} \geq 1, \forall m \in M, \forall k \in K
\end{aligned}
$$

There are two inter-dependent sub-problems concerning our objective to design *BoLTE*. The first sub-problem is to determine the optimal set of SFN clusters for each session. Based on this, the second sub-problem multiplexes different broadcast and unicast traffic flows on the spectrum resources of the eNBs.We formulate our PF utility maximization problem with the above Integer Program and list notations in Table I.

Output: Output variables are $X_{k,n}$, $Y_{m,n}$, $\mathbb{C}_k$, $F_{k,C}$, A_k.

Input: The input parameters are R_i, B, θ, $Q_{m,n}$, α_k (α_m).

Constraints: The first constraint ensures that, if the throughput of session k i.e., the achieved throughput of its net weakest subscriber is A_k, then the throughputs yielded by all other SFN clusters serving k (to their specific weakest UEs) is at least as high as A_k. The second and third constraints enforce that the scheduled data is limited by the data in the buffer. The fourth and fifth constraints indicate that, on any eNB, the total resource allocation for broadcast is limited by $\theta \cdot B$, and the net resource allocation for all flows is limited by B. The sixth constraint ensures integral allocation of resources. The last constraint ensures that at least one resource is allocated to any broadcast/unicast flow.

TABLE I
Notations

Symbol	Description
K	Set of broadcast sessions
M	Set of unicast UEs/flows
A_k	Throughput of broadcast session k
I	Set of entities (sessions and unicast flows), i.e., $I = K \cup M$
R_i	Finite buffer of entity i
$\mathbb{C}_k$	Set of SFN clusters configured to serve session k
C	An SFN cluster (i.e., a set of eNBs) $C \in \mathbb{C}_k$ serving session k
$F_{k,C}$	Rate yielded by SFN $C \in \mathbb{C}_k$ for session k
$Q_{m,n}$	Rate yielded by eNB n to unicast UE m
$X_{k,n}$	No. of resources allocated from eNB n for broadcast session k
$Y_{m,n}$	No. of resources allocated from eNB n to unicast UE m
$\alpha_{k(m)}$	Priority weight of broadcast session k (or unicast flow m)
B	Total number of resources available to each eNB per frame
θ	Broadcast resource budget on any eNB

Hardness: The optimization version of the NP-Hard multi-way partition problem [17] can be easily reduced to our version of the SFN clustering problem to generate a feasible set of non-overlapping SFN clusters for every session, covering all eNBs serving that session. We omit details for space constraints. The other sub-problem is to allocate resources across broadcast sessions, when the SFN clusters are determined. Even a simpler version of this problem is NP-hard [15].

IV. *BoLTE*: Solution Design

Given the hardness of our problem, we aim to design a lightweight, efficient and scalable solution that solves the two inter-dependent SFN clustering and resource allocation sub-problems. From Fig. 4, for a set of non-overlapping SFN clusters to be a feasible clustering solution for a session, the necessary conditions are:

1. Atleast one of the SFN clusters yields a higher common

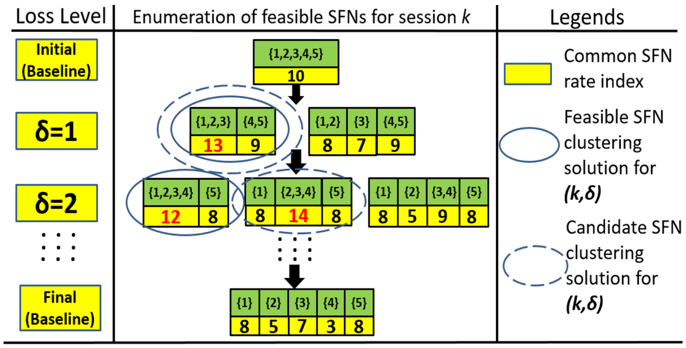


Fig. 5. Generation of candidate SFN clusters for session k.

rate than the single large SFN (SFN $\{1, 2, 3\}$ for K_1, and SFN clusters $\{2, 3\}$ and $\{4, 5\}$ for K_2 in Fig. 4(c)).
2. This happens only upon generation of smaller SFN clusters, where the net instantaneous weakest session subscriber gets a lower rate than the one corresponding to the single-large SFN (SFN clusters $\{4, 5\}$ and $\{1\}$ for K_1 and K_2, respectively).

This is a necessary but not the sufficient condition to determine the optimal set of session-specific SFN clusters. To determine the optimal set, it is required to enumerate the feasible sets of SFN clusters for all sessions and allocate resources from eNBs across all sessions/unicast flows based on the enumerated sets. This is illustrated in the hierarchy shown in Fig. 5, with reference to a given session k. The top level in the hierarchy shows the single-large SFN for session k yielding a common rate of index 10 for the net weakest session subscriber (also, happens to be the weakest UE served by the SFN). The next level in the hierarchy ($\delta = 1$) enumerates the feasible sets of SFN clusters, where the instantaneous net weakest session subscriber supports a rate indexed 1 less than the one yielded by the single-large SFN for the corresponding net weakest subscriber. Note that the instantaneous weakest subscribers for the same session k could vary between the single-large SFN and other clustering combinations. Similarly, any level δ in the hierarchy indicates that, for any feasible set of SFN clusters in this level, the instantaneous net weakest session subscriber supports a rate indexed δ less than the one from the single-large SFN. Fig. 5 depicts enumeration of feasible sets of SFN clusters for k meeting the above two conditions.However, there could be an exponential number of such feasible sets, and hence, resource multiplexing across sessions and unicast flows is combinatorial in nature. In the sub-sections below, we discuss heuristics to obtain the optimal set of SFN clusters and to efficiently allocate resources based on the SFN clustering solution, in polynomial time.

A. SFN Clustering

To generate feasible clusters, one can leverage graph representation of eNBs and utilize heuristics to the well-known minimum p-cut problem [5]. We construct a graph for each session, where vertices are eNBs (whose users subscribe to the given session) and weighted edges capture the transmit diversity gain between eNBs if they were to be synchronized in an SFN. Solving the min p-cut problem on this graph results in p-connected components, which correspond to SFN clusters for the session, with minimum inter-cluster interference among them. This is because the controlled loss in diversity gain (captured by the edge weight) for the net weakest session subscriber comes from removing the edges, creating smaller interfering clusters. The cut results in disconnected partitions, indicating non-overlapping SFN clusters. However, the min p-cut problem does not directly address our SFN clustering problem. Formally, min p-cut problem solves the following [5]: Given an undirected graph $G = (V, E)$ with positive real number edge weights and an integer p, find the set of edges $E' \subset E$ with minimum total weight whose removal creates p connected components in G. On the other hand, we aim to solve the following problem for SFN clustering: Given an undirected graph $G = (V, E)$ with positive real number edge weights, an integer p and an integer $\delta \in \mathbb{Z}^+$, create p connected components in G such that the rate of the net weakest session subscriber is at most δ worse than the one yielded by the single-large SFN (δ is as discussed earlier for Fig. 5). The parameter δ is crucial to our alternating optimization since it controls the amount of diversity gain traded off from multiplexing gain. We need to accordingly incorporate this in our graph construction to obtain the min-p-cut as a feasible solution to our problem.
1. We first construct a directed graph $G = (V, E)$ for any session k and loss level δ, i.e. tuple (k, δ), where each vertex $i \in V$ represents an eNB, and each edge $\{i, j\} \in E$ indicates that eNB i contributes an SINR of ρ_{ij} to j, where $\rho_{ij} \geq 0$.
2. To compute ρ_{ij}, we define x_j to be the SINR (in dB) of the instantaneous *weakest UE* (UE with the min. SINR) attached to eNB j, when we have a single large SFN. Now, if eNB i is removed from the single large SFN, it results in SINR degradation for eNB j's UEs since the diversity gain is reduced and interference is experienced from eNB i. Let us assume that the instantaneous weakest UE of eNB j (which could be different from the previous case) now has an SINR of $y_{i,j}$ dB.Thus, due to the impact of eNB i alone, eNB j loses $\rho_{ij} = x_j - y_{i,j}$ dB of SINR from the single large SFN.
3. We define MCS^k_{max} as the rate index of the net weakest subscriber of session k due to the single-large SFN (maximum diversity gain). For a given tuple (k, δ), we are trying to make sure that the rate of the resulting net weakest session subscriber is no less than $MCS^k_{max} - \delta$. Using a rate table mapping SINR values to encoding rate levels, we define $z_{k,\delta}$ to be the minimum SINR required to support a rate level of $MCS^k_{max} - \delta$. For each vertex j, we define weight $\hat{\rho}_j$ as $x_j - z_{k,\delta}$, capturing the maximum SINR margin that the eNB j affords to lose by moving away from a single large SFN.
4. Having defined the SINR margin ($\hat{\rho}_j$) of vertices and the impact (ρ_{ij}) between them, we now define the weight of any edge $\{i, j\}$ to be $w_{ij} = \log((1 - \min\{1, \frac{\rho_{ij}}{\hat{\rho}_j}\})^{-1})$. This captures the relative loss in SINR (i.e. $\frac{\rho_{ij}}{\hat{\rho}_j}$) at eNB j if eNB i were to be removed from its SFN cluster.

We use min-cut to remove edges with small weights and create multiple smaller SFN clusters out of the single large SFN, while still supporting a target rate level of at least $MCS^k_{max} - \delta$ for the instantaneous net weakest subscriber of session k. Note that larger edge weights denote significantly interfering eNBs, which would better be synchronized in an SFN (instead of cutting them apart) in order to prevent the rate of the weakest session subscriber from going below the target. The exponential choice of the weight function allows edge weights to go to ∞ when the vertex SINR margins $(1 - \min\{1, \frac{\rho_{ij}}{\hat{\rho}_j}\})$ approach 0, thereby preventing such edges from being cut. For larger values of δ (i.e., larger controlled loss in the net weakest subscriber rate), relative losses get smaller since the vertex SINR margins ($\hat{\rho}_j$) increase (while ρ_{ij} stays the same). Thus, we have the freedom to cut more edges (i.e., creating more SFN clusters that enable better multiplexing) as we progressively increase δ (i.e., decrease transmit diversity). Before we can use the min p-cut procedure on our graph, we need to convert it to an undirected graph. We do this by replacing the pair of directed edges $\{i, j\}$ and $\{j, i\}$ with a single undirected edge, with its weight being the sum of the pair of directed edges.

Algorithm 1 Discover max. utility: max_util

1: **for all** $k \in K$ **do**
2: Let MCS^k_{min} be the bottleneck rate of session k as a result of single-cell broadcast (i.e., min. diversity gain). Set $\delta := 0$, $\Delta_k = \emptyset$
3: **while** $MCS^k_{max} - \delta \geq MCS^k_{min}$ **do**
4: Let $\vec{\mathbb{C}}_k(\delta)$ be the vector containing candidate SFN clusters for session k and loss level δ
5: Set $\Delta_k := \Delta_k \cup \{\delta\}$, $\delta := \delta + 1$
6: **end while**
7: **end for**
8: Set $\Delta := \cup_{k \in K} \Delta_k$.
9: **for all** $\delta \in \Delta$ **do**
10: **for all** $k \in K$ **do**
11: Let $\xi(\mathbb{C})$ be the maximum common rate of any SFN in the set $\mathbb{C}$ of SFN clusters, $\forall \mathbb{C} \in \vec{\mathbb{C}}_k(\delta)$
12: Set $\hat{\mathbb{C}}_k(\delta) := \arg \max_{\mathbb{C} \in \vec{\mathbb{C}}_k(\delta)} \xi(\mathbb{C})$
13: **end for**
14: Let U_δ be the aggregate utility resulting upon allocating resources for the set I of entities, corresponding to δ.
15: **end for**
16: **for all** $k \in K$ **do**
17: Set $\overline{\delta} := \arg \max_{\delta \in \Delta} U_\delta$, and $\mathbb{C}_k := \hat{\mathbb{C}}_k(\overline{\delta})$.
18: **end for**
19: Return $\mathbb{C}_k$, $\forall k \in K$

After generating the un-directed graph as described above, we use a well-known heuristic to the min p-cut problem to generate p SFN clusters, where $p \in [2, |N|]$. It computes a min-cut, based on Ford-Fulkerson algorithm, in each of the $\overline{p} < p$ existing clusters by taking turns, while keeping the rest as such. That is, we get $\overline{p}$ sets of $\overline{p}+1$ new clusters. Among the resulting $\overline{p}$ sets of new $\overline{p}+1$ clusters, the set with the smallest cost (given by the sum of the weights of removed edges) is taken as the optimal. This process is repeated until p clusters are produced. Evidently, the algorithm increases the number of clusters by one at each iteration. It has an approximation factor of $2(1 - \frac{1}{p})$ [5]. Out of the resulting $p - 1$ sets of p SFN clusters for the tuple (k, δ), we short-list the ones that satisfying the two necessary conditions for feasibility: (i) the resulting net weakest session subscriber must support a rate of at least $MCS^k_{max} - \delta$, and (ii) at least one constituent SFN cluster must yield a common rate of MCS^k_{max}.

In Fig. 5, since the single large SFN yields a common rate index of 10, for $\delta = 1$, the target rate index is 9 (i.e. 10-1). The set $\{\{1, 2, 3\}, \{4, 5\}\}$ is a feasible set since the common SFN rate level yielded by $\{1, 2, 3\}$ is 13 (greater than 10 from the single-large SFN) and $\{4, 5\}$ level is 9, which satisfies the target level for the resulting net weakest session subscriber. But the SFN $\{3\}$ in the set $\{\{1, 2\}, \{3\}, \{4, 5\}\}$ yields a bottleneck rate level of 7, that is less than the target level. So, this set is not a feasible solution. Similarly, for $\delta = 2$, $\{\{1, 2, 3, 4\}, \{5\}\}$ and $\{\{1\}, \{2, 3, 4\}, \{5\}\}$ are feasible sets. Upon enumeration, the procedure returns $\vec{\mathbb{C}}_k(\delta)$, which is a vector containing the feasible SFN clustering solutions for all feasible values of p.

We describe how we determine the optimal solution from the vector of feasible solutions for (k, δ). We integrate the SFN clustering procedure with resource allocation to discover the maximum utility, as seen in Algorithm 1. For any session k, we exhaust all values of δ, starting from the one corresponding to single-large SFN (where $\delta = 0$) till the one corresponding to single-cell broadcast, and generate a vector of feasible solutions for the session. (Lines 3-6). To avoid the resulting exponential complexity from the combinatorial nature of resource multiplexing across sessions considering all feasible SFN clustering solutions, we decouple clustering from resource allocation but pick a clustering solution intelligently (with $O(|K|)$ complexity), contributing to better multiplexing gains from resource allocation.

Candidate solutions: Among all feasible solutions in the vector $\vec{\mathbb{C}_k}(\delta)$, we select that set of SFN clusters which has the maximum common rate for any constituent SFN (Lines 11-12), since it would benefit the most from being separated from the single-large SFN cluster in terms of supporting higher rates. This is highlighted in Fig. 5. We call it the candidate solution $\hat{\mathbb{C}}_k(\delta)$. Once we find the candidate SFN solution for each tuple (k, δ), we repeatedly run resource allocation on the candidate solutions across all sessions/unicast flows, as detailed in Section IV-B, and for all δ values (Line 14). This way, we can observe the impact of giving up diversity gain (by increasing δ) depending on how resources from the eNBs are multiplexed across all flows. By recording the utility U_δ produced with each δ, we then finalize the solution that corresponds to the maximum utility (Lines 17-18).

B. Resource Allocation

We now describe our resource allocation strategy based on a water-filling (WF) variant of the PF algorithm [14], [15] across sessions/unicast flows, for a given δ.

1. First, we allocate a unit resource to each flow (broadcast/unicast) based on $\hat{\mathbb{C}}_k(\delta)$ for the tuple (k, δ). Set $X_{k,n} = 1$, $\forall n \in C \in \hat{\mathbb{C}}_k(\delta)$ and $\forall k \in K$. Set $Y_{m,n_m} = 1$, $\forall m \in M$, where $n_m \in N$ is the eNB to which unicast UE m is attached. Note that when allocating for a broadcast flow, the same resource is given to all the eNBs that are in each SFN cluster contained in $\hat{\mathbb{C}}_k(\delta)$.

2. Next, the algorithm discovers the flow that yields the maximum PF utility to allocate the next unit resource. For PF, the marginal utility of any flow i corresponds to the weighted

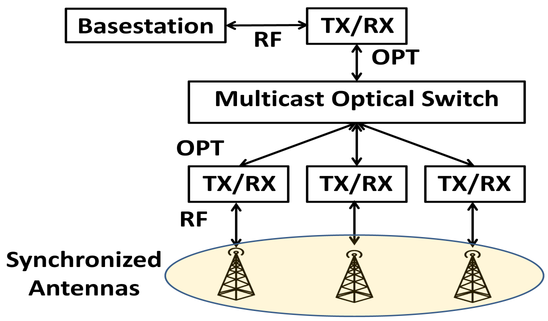


Fig. 6. We create SFNs by virtue of an optical switch.

ratio of the residual achievable throughput $\overline{R}_i$ of the flow to its past-achieved throughput A_i.

3. Let us assume that the flow with the maximum metric (denoted by i') is a broadcast session. In this case, increment $X_{k,n}$ by 1 and set $A_{i'} = \min_{C \in \hat{\mathbb{C}}_{i'}(\delta)} \{F_{i',C}.X_{i',n}\}$, $\forall n \in C$, as defined in Section II. We then determine the SFN with the minimum common rate as C' (i.e. $C' = \arg \min_{C \in \hat{\mathbb{C}}'_i(\delta)} \{F_{i',C}\}$) and allocate one resource from all eNBs in C'. If the net throughput yielded by C' becomes greater than that from any other SFN C, then we allocate one resource from all eNBs, synchronized in C. Recall that the achieved session throughput varies with different SFN clustering combinations due to variation in the instantaneous bottleneck UE of the SFN.

4. We then determine the session throughput as the minimum achieved of all SFN throughputs. The SFN cluster containing the net weakest session subscriber allocates the most resources for the session than the rest. Note that this allocation is subject to broadcast budget and bit-rate constraints.

5. On the other hand, if the flow with the maximum PF metric i.e. i' is for a unicast UE, then a unit resource is allocated from the eNB serving i', given by $n_{i'}$. In this case, increment $Y_{i',n_{i'}}$ by 1, and set $A_{i'} = Q_{i',n_{i'}}.Y_{i',n_{i'}}$. For unicast flows, the net resource allocation should be limited to B resources on any eNB, along with bit-rate constraints.

6. Any flow violating the constraints is removed from the set and its achieved throughput (A_k for session k, A_m for UE m) is noted. The resource allocation process is repeated until all flows violate the constraints.

7. Once the achieved throughputs of all flows are computed, the net aggregate system utility for each δ is computed as $U_\delta = \sum_{i \in I} \alpha_i \log A_i$ and returned to Line 14 of Algorithm 1.

The overall complexity of the joint algorithm is $O(|\Delta|(|K|S_{cut}(|N|) + S_{alloc}))$, where $S_{cut}(|N|)$ is the complexity of the min p-cut heuristic $O(p.|N|^3)$, when run for $p = |N|$, and S_{alloc} is the complexity of our resource allocation procedure $O(B|K||N|)$.

V. Performance Evaluation

A. Implementation and Testbed

Since eMBMS itself is a very recent technology, to the best of our knowledge, there are no experimental platforms with LTE supporting SFNs. To overcome this limitation, we build an SFN testbed using an RF-over-Fiber (RFoF) based C-RAN system with WiMAX base stations as RF sources [13], [16]. Although WiMAX and LTE are different standards, they are both based on the same OFDMA technology with similarity in the notion of resource allocation in a 2D frame and in the impact of SFNs. The components of our testbed are: (i) *a server* to stream broadcast videos using VLC and unicast traffic using iPerf, (ii) *three WiMAX base stations* flashed with schedulers written in C and built on Fedora 12 OS, used as RF sources, (iii) *an optical switch* used to create synchronized SFNs, (iv) *three remote radio heads (RRH) or antennas* - to transmit RF, and (v) *commercial WiMAX UEs*.

Enabling the SFNs: We take the RF output of a base station and convert the analog RF signal to an optical signal. This signal is then fed to a multicast optical switch that can be programmed to map the input signals to one or more of its output ports. After passing through the switch, each optical signal is converted back to RF before being transmitted over-the-air by RRHs. When the switch is programmed to replicate one input signal on multiple output ports, each RRH receives the same optical signal (as shown in Fig. 6). Hence, with the conversion to RF and over-the-air transmission, we effectively have multiple RRHs transmitting the same RF signal coming from one base station in a synchronized manner as in an SFN. By virtue of the optical switch and the BBU-RRH logical mapping, we can enable very tightly synchronized SFNs without the need for an explicit synchronization protocol across RRHs. Being software programmable, the switch enables us to realize clusters ranging from single-cell broadcast to a single-large SFN and other combinations in between. It has PHY/MAC functionalities enabling modulation rate adaptation, scheduling and our custom modules for multicast support.

We implement our algorithms on the scheduler, connected to the optical switch via RS-232 and interfacing with each base station over Ethernet. Based on the SFN clusters determined by our algorithm, the PC configures the switch and the scheduler allocates the determined rate and resources.

B. Experimental Evaluation

We deploy 3 adequately-interfering RRHs $\{\mathcal{C}, \mathcal{I}, \mathcal{K}\}$ (connected to BBUs) in a 50mx50m floor, emulating our SFN. We position UEs at 40 randomly selected locations in our floor to generate different channel signatures from the RRHs. We stream 2 1080p broadcast sessions K_1 and K_2 using VLC that supports Variable Bit Rate (VBR) streaming over H.264, in addition to unicast data streaming using iPerf. We evaluate the following metrics in our experiments: (i) *broadcast spectrum utilization (Minimize)* - defined as the total fraction of the resources allocated in a frame across all eNBs serving the two broadcast sessions, (ii) *aggregate broadcast throughput (Maximize)* of the two sessions, (iii) *aggregate unicast throughput (Maximize)* of all the UEs with unicast flows, and (iv) *received video rate (Maximize)* of the streamed session at the weakest subscriber, indicating resolution.

In the interest of space, we present our evaluation for two topologies. The two topologies indicate two different sets of user locations, yielding different channel conditions for the UEs (especially the weakest), that further results in different

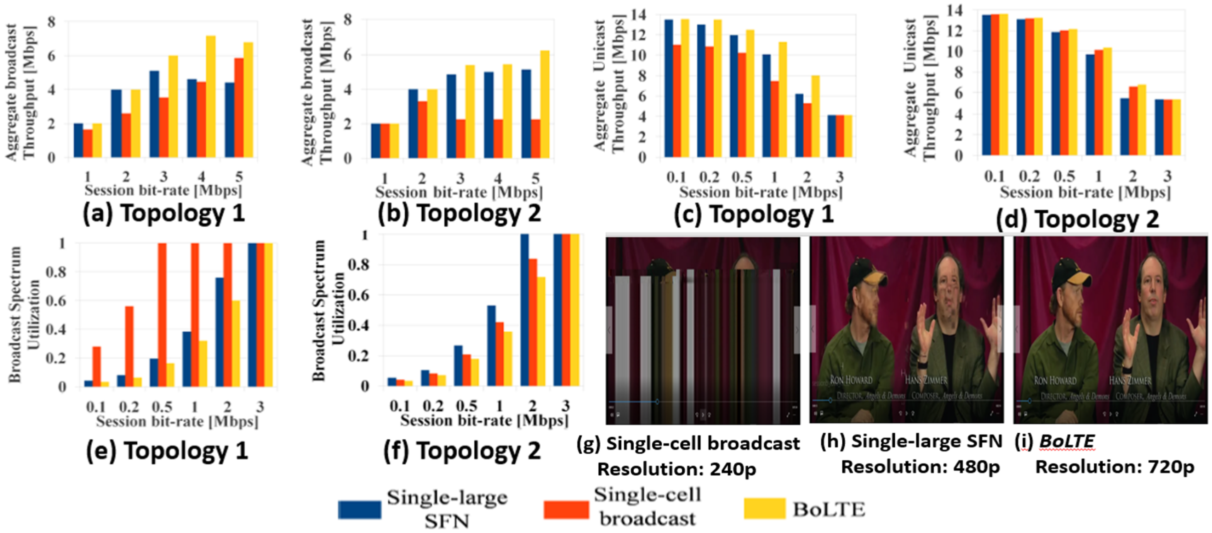


Fig. 7. Experimental evaluation of *BoLTE* using a variety of performance metrics (Legends are at the bottom)

optimal SFN clustering combinations. Our observations in other topologies are consistent with the results presented here. Since the UE distribution varies between the topologies, *BoLTE* yields a different SFN clustering solution for each of the topologies. For topology 1, *BoLTE* produces the clustering solution $\{\{\mathcal{C},\mathcal{K}\},\{\mathcal{I}\}\}$ for both K_1 and K_2. For topology 2, it yields $\{\{\mathcal{I},\mathcal{C}\},\{\mathcal{K}\}\}$ as the clustering solution for both sessions (we observed different clustering solutions across sessions in other topologies). We carry out experiments by varying each broadcast session's bit-rate from 0.1 to 5 Mbps and evaluate the above metrics by comparing *BoLTE* with single-cell broadcast and single-large SFN baselines in Fig 7.

Broadcast Throughput: In Figures 7(a) and (b), we present the performance of the aggregate broadcast throughput for broadcast sessions with bit-rates from 1 Mbps to 5 Mbps. For lower bit-rates (i.e., 1 Mbps and lower), we observe that all schemes can mostly satisfy the demand of both the sessions for a total throughput of 2 Mbps (albeit with different resource utilizations). When sessions have higher bit-rates, efficiently utilizing resources becomes more important since there is a finite number of resources available for broadcast. Interestingly, we observe that single-large SFN outperforms single-cell broadcast for low bit-rates but this trend is reversed for high bit-rates (e.g., 5 Mbps) in topology 1. For lower bit-rates, the impact of diversity gain is more pronounced since it can offer higher rates. and satisfy both sessions' demands fully (i.e., for a total of 4 Mbps when per-session rate is 2 Mbps). On the other hand, single-cell broadcast suffers from lower rate due to interference and cannot satisfy the demand within the fixed broadcast budget. For high bit-rates however, even the higher rate enabled by single-large SFN is not sufficient to satisfy the demand within the resource budget. In this regime, it is more critical to multiplex resources across multiple sessions. By appropriately accounting for this tradeoff, we see that *BoLTE* consistently outperforms baseline schemes (by max. 100% for bit-rates larger than 4 Mbps).

Unicast Throughput: We now present the results for aggregate unicast throughput, but now, by varying the session bit-rates from 100 Kbps to 3 Mbps in Figs. 7(c) and (d). Since broadcast bit-rates are lower, the impact of resource allocation will be more on unicast flows. For topology 1, we observe that *BoLTE* yields better unicast throughput than both baselines (except for 3 Mbps session rate). The increased unicast throughput is owing to better utilization by *BoLTE* (achieved via intelligent SFN clustering) for broadcast sessions. Since broadcast sessions occupy less resources with *BoLTE* (lower than the broadcast budget of $\theta \cdot B$), there are more resources left for unicast flows. Indeed, the trend of broadcast utilization ratio metric in Figure 7(e) confirms this hypothesis (lower broadcast utilization yields better unicast throughput). Note also that all schemes have full utilization of 1 for broadcast when the session rate is 3 Mbps, explaining the same unicast throughput yielded by all schemes in Figure 7(c). Also, due to smaller spectrum utilization gains for topology 2 (as in Fig. 7(f)), the unicast gains for *BoLTE* in Fig. 7(d) are modest.

Video rate: Figs. 7(g,h,i) demonstrate the quality of the received video streams at the instantaneous net weakest subscriber for session K_2. The single-cell broadcast yields a rate index of 1, an allocation of only 6 resources, resulting in a video rate of 404 kbps, due to loss of diversity/multiplexing gains. The corresponding video frame shown in Fig. 7(g) is highly lossy. The single-large SFN yields a rate index of 3, allocates 60 resources, resulting in a 1.44 Mbps video rate. This is due to higher diversity gains. *BoLTE* gets adequate diversity and multiplexing gains, resulting in a rate index of 2 but allocation of 135 resources, corresponding to 3.87 Mbps.

C. Large-scale Simulations

Here, we evaluate the performance of *BoLTE* using large-scale simulations for: (i) the aggregate broadcast throughput of all sessions, (ii) the aggregate unicast throughput of all UEs, (iii) aggregate throughput, defined as the sum of the throughput of all broadcast sessions, scaled by the number of session subscribers, and unicast flows. Since a broadcast session is limited by the bottleneck UE in its eMBMS group, average broadcast session throughput is significantly less

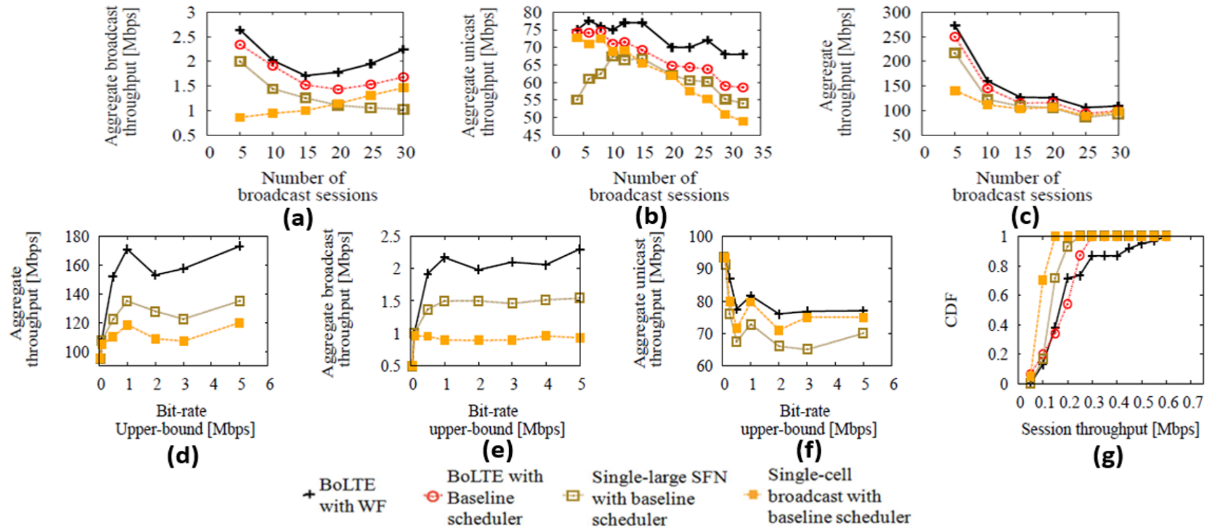


Fig. 8. Evaluating SFN clustering and resource allocation components in *BoLTE* (Legends are at the bottom)

than the average unicast throughput. We scale the broadcast throughput (only when presenting aggregate throughput) to account for this disparity and improve plot clarity. (iv) CDF of the broadcast throughput showing the PF based on Eqn. (1). We use a custom LTE simulator conforming to 3GPP Release 8/9 standards, integrated with simulation traces using the NS3 open-source LTE/EPC network simulator. The topology consists of 25 uniformly distributed eNBs (with transmission power of 23 dBm) and 500 randomly distributed UEs. Each UE subscribes to either one broadcast session or has a unicast flow. Each eNB has a total of 500 resources in a frame and the broadcast budget is 300. Each result is an average of 50 trials, varying UE placement and traffic subscription by UEs. We analyze the gains yielded individually and collectively by SFN clustering and the WF resource allocation in *BoLTE*. For SFN clustering gains, we evaluate *BoLTE* against baseline schemes of single-large SFN and single-cell broadcast. To evaluate the gains from WF resource allocation, we evaluate it against a baseline PF scheduler, which allocates resources to a flow in proportion to the ratio of its demand to its throughput.

Figure 8(a) plots the aggregate broadcast throughput by varying the number of broadcast sessions (with each session bit-rate at 10 Mbps). We see an interesting cross-over point, where single-large SFN outperforms single-cell broadcast for less number of sessions but the trend is reversed as we increase the number of sessions. This is because with fewer sessions, there are more subscribers per session. With a large subscriber pool, the likelihood of having a large variance between the rates supported by each UE increases for each session. With higher cell-edge UEs, the performance of the session via single-cell broadcast suffers from inter-cell interference. Hence, the notion of SFN greatly benefits the broadcast performance in this regime by increasing the diversity gain and the common rate. For a larger number of sessions, there are fewer UEs with poorer rates per session. In this regime, broadcast performance benefits more from the ability of eNBs to independently select rates and multiplex resources across different flows. On the other hand, we observe significant gains with *BoLTE* (as much as 150%) owing to properly addressing the tradeoff between diversity gain and resource multiplexing gain. When we analyze the gains from each component, we see that most of the gain comes from *BoLTE*'s SFN clustering when we have fewer sessions (less gap between WF and baseline scheduler). The contribution of the WF scheduler (due to its fairness) increases with more number of sessions. Fig. 8(b) plots aggregate unicast throughput by varying the number of broadcast sessions. Since the single-large SFN enforces common rate and resource allocation across all eNBs and diversity gain is dominant with fewer broadcast sessions, there are comparatively less number of remaining resources for unicast. Thus, the single-large SFN yields lower unicast throughput than single-cell broadcast for fewer sessions. On the other hand, due to a greater need to multiplex resources across sessions for higher number of broadcast sessions, fewer resources are scheduled for unicast. Thus, single-cell broadcast yields lower unicast throughput than single-large SFN for higher number of sessions. *BoLTE* consistently exhibits up to 50% gains in aggregate unicast throughput over other scheme, due to gains from the WF scheduler. Similarly, we see *BoLTE*'s increased overall throughput by max. 80% in Fig. 8(c).

Figures 8 (d), (e), and (f) plot the aggregate throughput, aggregate broadcast throughput and aggregate unicast throughput by varying the per-session bit-rate (number of broadcast sessions is fixed to 10). For lower bit-rates, all schemes yield similar broadcast throughput satisfying all broadcast demand. As bit-rate increases, we see that single-large SFN outperforms single-cell broadcast due to similar reasons as in Fig. 8(a) (since having 10 sessions creates the regime where diversity gain is more pronounced). For high bit-rates, all schemes saturate their broadcast budget and hence cannot further increase broadcast throughput. We observe that *BoLTE* consistently outperforms the baselines with gains as much as 100%. Again similar to our earlier observations, the trend for unicast throughput is opposite to that of broadcast. To complement our aggregate results, we analyze the fairness of our WF scheduler by plotting the CDF of the session throughput in Fig. 8 (g) (for 10 broadcast sessions with 1 Mbps bit-rate). We see that with *BoLTE*, a larger number of broadcast sessions get higher throughputs, indicating a high PF utility given in Eqn. (1). Thus, individual session throughputs are not sacrificed. While the maximum per-session throughput by baseline schemes is 275 Kbps, *BoLTE* yields 600 Kbps.

VI. Related Work

Some of the fundamental aspects of SFN transmissions in terms of rate selection and resource scheduling are discussed in [1], [10]. However, they do not discuss the aspects of SFN clustering, accounting for the diversity trade-offs in the system. The trade-off between transmission diversity and transmission power in improving the spectrum efficiency of an SFN system is discussed in [3]. While our paper deeply analyzes the impact of transmission diversity, it does not modify the transmission power but closely looks at the inter-dependent factor of resource multiplexing. One of the most recent studies on eMBMS is by [6], where the authors address scheduling by grouping users with similar channel conditions. Hence, they aim to minimize the encoding rate discrepancy across broadcast subscribers while retaining the transmission efficiency of broadcast. The main difference between this work and ours lies in the definition of broadcast session performance and the nature of SFN clustering. While we define the broadcast session performance as the minimum of the throughputs achieved by its network-wide subscribers and maximize the aggregate performance of all sessions in our optimization, their work does not define this quantity. Instead, they partition the subscribers into different eMBMS groups and their objective is to maximize the aggregate utility of the groups. This is because their work does not consider multiple broadcast sessions and assumes only one SFN area, unlike our work. For this, they group UEs with similar channel conditions. They sort the UEs in the network based on channel conditions and use dynamic programming to form eMBMS groups such that they maximize their objective function. While we too partition the eNBs (and hence, their UEs) into different SFN clusters (and hence, eMBMS groups) per session, we manage the diversity-multiplexing trade-off in forming the clusters, so that we maximize the aggregate performance of all sessions, while giving up on the per session diversity gain for its weakest subscriber. Another study focuses on cooperating transmission for broadcast sessions in relay networks [15]. The authors focus on optimizing the resource allocation given a set of pre-determined (static) SFNs. They do not explore the more challenging problem of jointly optimizing SFN clusters and resource allocation, as we consider in our work. In [11], the authors evaluate the effect of the SFN size (in terms of number of eNBs) on eMBMS services such as multicast video and unicast traffic but they do not propose any solutions for determining SFNs and scheduling resources. The authors in [1] evaluate different rate adaptation mechanisms for eMBMS using simulations but do not propose any solution related to our problem. Other studies such as [12], [16] focus on rate adaptation for multicast video delivery in LTE without the notion of SFN. SFN area formation and clustering based on application content similarity are considered in [4].

VII. Conclusions

We introduced *BoLTE*, an efficient multi-cell broadcasting framework using the notion of *single frequency network*, for eMBMS operations in LTE. *BoLTE* dealt with efficiently allocating radio resources across LTE eNBs to serve multiple broadcast sessions and unicast traffic. *BoLTE* accounted for two important factors in its design: (i) Transmit diversity gain, leveraging cooperation among eNBs to improve SINR, and (ii) Resource multiplexing gain, leveraging channel diversity among UEs.We analyzed the trade-off between these two factors and developed algorithms for (i) clustering eNBs into session-specific SFNs, and (ii) allocating resources for eMBMS groups and unicast users, that synergistically captured this trade-off. We implemented a prototype of *BoLTE* on our experimental testbed and evaluated *BoLTE* extensively using simulations to show improvements.

References

[1] A. Alexiou, C. Bouras, V. Kokkinos, A. Papazios, and G. Tsichritzis. Efficient MCS selection for MBSFN transmissions over LTE networks. In *Proc. IFIP Wireless Days*, oct 2010.

[2] 3GPP TS 23.246: Multimedia Broadcast/Multicast Service (MBMS): Architecture and Functional Description. http://www.3gpp.org/DynaReport/23246.htm.

[3] R. E. Bettancourt and J. M. Peha. On the trade-off between Spectrum Efficiency and Transmission cost in traditional and SFN-based Broadcast Television. In *Proc. IEEE DySPAN*, September 2015.

[4] C. Borgiattino, C. Casetti, C. F. Chiasserini, and F. Malandrino. Efficient area formation for LTE broadcasting. In *Proc. IEEE SECON*, June 2015.

[5] G. Calinescu, H. Karloff, and Y. Rabani. An Improved Approximation Algorithm for multi-way cut. *Elsevier, Journal of Computer and System sciences*, 60(03), June 2000.

[6] J. Chen, M. Chiang, J. J. Erman, G. Li, K. K. Ramakrishnan, and R. Sinha. Fair and optimal resource allocation for lte multicast (embms): Group partitioning and dynamics. In *Proc. IEEE INFOCOM*, apr 2015.

[7] M. Eriksson, S. M. Hasibur Rahman, F. Fraille, and M. Sjostrom. Efficient Interactive Multicast over DVB-T2 - Utilizing Dynamic SFNs and PARPs. In *Proc. Int'l Conf. Computer and Information Technology (BMSB)*, June 2013.

[8] A. Jalali, R. Padovani, and R. Pankaj. Data throughput of CDMA-HDR, a high efficiency-high data rate personal communication wireless system. In *Proc. IEEE VTC*, aug 2000.

[9] LTE Broadcast: Evolving and Going Beyond Mobile. https://www.qualcomm.com/documents/lte-broadcast-evolving-and-going-beyond-mobile.

[10] S. Lu, Y. Cai, L. Zhang, J. Li, P. Skov, C. Wang, and Z. He. Channel-aware frequency domain packet scheduling for MBMS in LTE. In *Proc. IEEE VTC*, April 2009.

[11] J. Monserrat, J. Calabuig, A. Fernandez-Aguilella, and D. G. Barquero. Joint delivery of unicast and eMBMS services in LTE networks. *IEEE Trans. Broadcasting*, 58(02), 2012.

[12] S. Sharangi, R. Krishnamurthi, and M. Hefeeda. Energy-efficient multicasting of scalable video streams over WiMAX networks. *IEEE Trans. Multimedia*, 13(01), February 2011.

[13] K. Sundaresan, M. Y. Arslan, S. Singh, S. Rangarajan, and S. V. Krishnamurthy. FluidNet: a flexible cloud-based radio access network for small cells. In *Proc. ACM MobiCom*, 2013.

[14] K. Sundaresan and S. Rangarajan. Scheduling algorithms for video multicasting with channel diversity in wireless ofdma networks. In *Proc. ACM Mobihoc*, may 2011.

[15] K. Sundaresan and S. Rangarajan. Cooperation versus multiplexing: Multicast scheduling algorithms for ofdma relay networks. *IEEE/ACM Trans. Networking*, 22(03), 2014.

[16] J. Yoon, H. Zhang, S. Banerjee, and S. Rangarajan. Muvi : A multicast video delivery scheme for 4G cellular networks. In *Proc. ACM Mobicom*, sep 2012.

[17] L. Zhao, H. Nagamochi, and T. Ibaraki. Greedy-splitting algorithms for approximating multiway partition problems. *Springer Mathematical Programming*, 102(01), January 2005.