EURECOM
Communication Systems Department
Campus SophiaTech
CS 50193
06904 Sophia Antipolis cedex
FRANCE

Research Report RR-17-332

QoS Guarantee in Self-Backhauled LTE Mesh Networks

April 18^{th}, 2017

Romain Favraud, Navid Nikaein and Chia-Yu Chang

Tel : (+33) 4 93 00 81 00
Fax : (+33) 4 93 00 82 00
Email : {romain.favraud,navid.nikaein,chia-yu.chang}@eurecom.fr

[1]EURECOM's research is partially supported by its industrial members: BMW Group Research and Technology, IABG, Monaco Telecom, Orange, Principaut de Monaco, SAP, ST Microelectronics, Symantec.

QoS Guarantee in Self-Backhauled LTE Mesh Networks

Romain Favraud, Navid Nikaein and Chia-Yu Chang

Abstract

LTE is deployed in most countries and continuously evolving to match new user requirements. While it is expected to support 5G deployments for outdoor and long range communications, it will also be used for Public Safety services in major countries. Among that, new scenarios call for wider networks on the move relying on dynamic meshing of the base stations. Leveraging the LTE air interface for base station meshing is an appealing idea but is not that straight forward to ensure high performances. In this article we study and evaluate scheduling strategies for multi-hop LTE mesh networks relying on the LTE relay channel. We first present the LTE relay channel and the scheduling problems of in-band LTE backhauling. We then propose a practical cross-layer method for resource allocation in order to meet QoS requirements of specific flows in such a network. We finally evaluate our proposed method in realistic scenarios. We show the effectiveness of the approach compared to a legacy method for meeting QoS requirements of real-time flows while improving throughput of elastic flows.

Index Terms

LTE, Relay Physical Channels, Self-backhauling, Mesh Networks, Coordinated Scheduling, QoS specific approach.

Contents

List of Figures

1 Introduction

Long Term Evolution (LTE) is now the 4G cellular network of reference. 3GPP LTE specifications are expanding its use-cases and increasing its features at every new release in order to become the 5G outdoor and long range radio access technology (RAT). Moreover, LTE is expected to be the next RAT for Public Safety (PS) communications and specifications items have emerged in this regard [1]. PS networks have several specific requirements including reliability and resiliency which demand specific Quality of Service (QoS) to be addressed. In common PS scenarios, LTE Base Stations (BSs), called eNodeBs (eNBs), may lose access to the Evolved Packet Core (EPC) due to some outage or specific deployments. When that happens, they are not anymore able to provide any service to their served users. This is an issue for PS networks and is addressed by 3GPP through the Isolated E-UTRAN concept that allows eNBs to continue providing minimal services for local PS users (TS 22.346, TR 23.797). In [2], we proposed an evolution of the Isolated E-UTRAN concept in the form of a new BS architecture for nomadic LTE networks that we called an e2NB. It embeds essential core network functions to ensure local services and has the ability to connect to other similar BSs leveraging the LTE relay interface so as to create a mesh network. Fig. 1 shows an example network topology. The e2NB allows supporting new use cases where dynamic meshing among fix and/or moving BSs is highly required. In [3], we showed that the relay channel (called *Un*) performance is close to the legacy LTE eNB ↔ User Equipment (UE) interface (called *Uu*) and allows for efficient transmissions between e2NBs. However, one of the main challenges when meshing e2NBs, especially with single frequency reuse, is that the transmissions/receptions and resource allocation between the BSs have to be coordinated to avoid excessive blocking resulting from interference and guarantee performance of end-to-end connections.

In this paper, we show how the relay channels can be leveraged and coupled with a smart resource allocation policy to evolve LTE from a single-hop network to a multi-hop meshing of several fixed and/or mobile base stations. To this end,

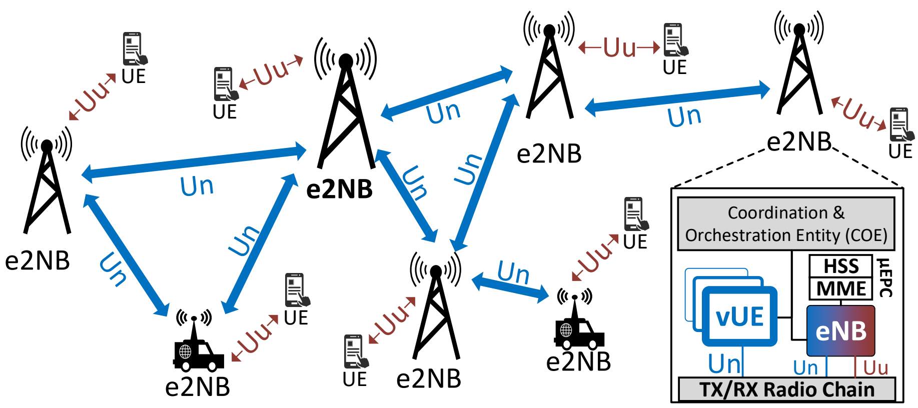


Figure 1: LTE mesh network based on e2NB with LTE backhaul

we briefly recall the properties of the LTE relay physical channels and the problems of single frequency wireless mesh networks in Section 2. In Section 3, we first present our target use cases and then state the associated problems, objectives and variables. In Section 4, we propose a cross-layer semi-centralized method to allocate resources on the network while taking into account QoS requirements. Finally, in Section 5, we evaluate the performance of the proposed solution over several topologies using a simulator we developed on a realistic scenario and we compare the results to other resource allocation algorithms. We conclude this work based on these results and present the next steps of our research.

2 Background on in-band LTE backhauling and scheduling problems

The proposed solution is based on the use of the LTE relay channel for in-band self-backhauling (i.e. single frequency networking) to mesh LTE BSs.

2.1 LTE relay channel (*Un*)

LTE Relay is specified in 3GPP Release 10 allowing a relay node (RN) to serve UEs on its access link and reach the EPC through its backhaul link with its anchor eNB, called Donor eNB (DeNB). The LTE relay interface is denoted as *Un* [4] and is used to realize the in-band backhaul link between a LTE RN and its DeNB. As this link is in-band, it shares the same carrier frequency as the access link relying on a time division multiplexing mechanism. To ensure that the channel between the RN and its UEs complies with the legacy access interface (*Uu*), the *Un* interface relies on specific properties of a mechanism originally introduced in LTE eMBMS (enhanced Multimedia Broadcast Multicast Service, a.k.a. MBSFN) to differentiate multicast/broadcast subframes from the unicast ones. We recall that LTE frame is structured into ten 1ms subframes (SFs) that carries a number of resource blocks (RBs) that depends on the channel bandwidth. In a LTE FDD (Frequency Division Duplex) frame, a maximum of 6 MBSFN SFs are allowed. A relay can exploit properties of these MBSFN SFs to receive specifically formatted unicast RBs from its DeNB on the downlink (DL) channel instead of always being transmitting to its UEs on the access link [4]. This means that the maximum data rate on the *Un* interface is reduced to 60% of what can be achieved on the *Uu* interface.

2.2 Single frequency wireless mesh network

While there are several works regarding the use of LTE *Un* interface for self-backhauling [5] [6], to the best of our knowledge, none of them considered meshing BSs that host their own EPC and that do not have a unique backhaul path towards a gateway. As indicated hereinbefore, we advocate that the *Un* interface can be re-used to extend the eNB capability with in-band self-backhauling and realize self-organized LTE mesh network among the BSs [1] [2]. The e2NB incorporates virtual UEs (vUEs) that are used to initiate and maintain connections to the adjacent e2NBs using the *Un* interface.

Using the *Un* interface to realize the backhaul of a LTE network in a mesh fashion (as shown on Fig. 1) is very similar to a TDMA based wireless mesh network. However, realizing an efficient wireless mesh network on a single frequency is still an open research problem. As all nodes share the same frequency resources, each transmitter becomes a potential interferer. Thus, collisions among transmissions may happen if the channel access is not coordinated or canceled, which in turn limits the achievable rate. There exist many issues in wireless mesh networks as surveyed in [7], including (a) topology control, (b) routing, (c) link scheduling, (d) interference measurement and (e) power control. The aforementioned problems are highly inter-dependent across layers and transmitting nodes.

To deal with these issues jointly, we propose a coordinated and cross-layer approach to unlesh the performance barriers when meshing e2NBs. In this paper, we aim to guarantee the QoS in per-flow basis of a self-backhauled LTE mesh network by jointly considering the topology control, routing, link scheduling, and power control. The considered architecture is based on a logically centralized control entity that manages and orchestrates the induced mesh network across medium access control (MAC) and network layer through *policy enforcement*, as described in [1] [2].

3 Problem statement and objectives

As mentioned in the previous section, it is necessary to define and enforce policies spanning across the relevant network layers and nodes to realize an efficient single frequency mesh network. We first define the objectives considering the use cases and then present the problem as well as its variables.

3.1 Traffic flows

In this paper, we focus on public safety scenarios with two types of traffic flow: real-time traffic with latency requirement (e.g. VoIP) and elastic traffic that can be treated as best effort. The considered scenario extends the Isolated E-UTRAN operation in that each BS can not only operate independently and host its own services, but also provide services to the other BSs through the self-bakhauled LTE mesh network. The traffic pattern in this scenario is heterogeneous and can be composed of intra-cell and inter-cell UE-to-UE (U2U), eNB-to-eNB (e2e) and UE-to-eNB (U2e) communications. For the real-time traffic, we consider VoIP traffic with the target maximum one-way-delay of 150ms for 95% of the packet to ensure a quality call with a Mean Opinion Score (MOS) of 3.5 using a G.729 codec [8].

For the elastic traffic, we consider multimedia data transfer with the objective of maximizing the requested throughput.

3.2 Problem statement

We assume a single frequency FDD LTE network relying on e2NBs [3]. Each e2NB is equipped with a single omni directional antenna. Based on two entities (eNB, vUE(s)) and two interfaces (Un, Uu) in e2NB (see Fig. 1), a DL and UL SFs allocation to either entity with corresponding interface is illustrated in Fig. 2 using all SFs (6 MBSFN SFs and 4 non-MBSFN SFs).

In our problem, we aim to allocate the (a) MBSFN SFs for backhaul transportation, (b) destination e2NB of backhaul relaying and (c) transmission power. The resource allocation algorithm enforces policy to each e2NB on allocated TX and RX SFs. This policy depends on the traffic patterns on the mesh (which is a combination of U2U, U2e and e2e communications), on the diverse objectives (latency requirements, throughput maximization, etc.) of higher-layer applications, on the network topology, and on the choices of routing paths. Therefore, the main variables at the network level are:

- TX/RX SF allocation policy at each eNB
- Power allocations for each SF and at each eNB
- Forwarding paths/tables

During the allocation, a trade-off can be observed between allowing more e2NBs to communicate while in the mean time reducing the perceived interference by each node. Considering the two types of traffic flow introduced in the previous subsection, our approach has two objectives:

- Guarantee the latency of real-time flows (i.e. VoIP)
- Maximize the throughput of elastic flows

4 Coordinated scheduling

The proposed approach is based on a logically centralized coordinated and orchestration entity (COE) that schedules nodes and selects data forwarding paths by enforcing a policy to each BS (refer to Fig. 1). The COE follows a hierarchical

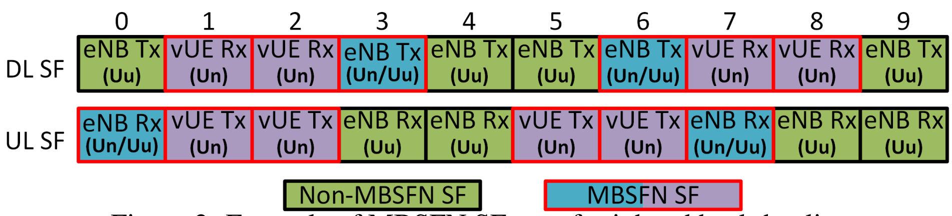


Figure 2: Example of MBSFN SFs use for inband back-hauling

design and is composed of a centralized entity that is connected to a number of COE agents [9], one per e2NB in a typical case. The COE agent can either act as a local COE with a limited network view handling control delegated by the centralized entity, or in coordination with other agents and the centralized entity. The communication between the centralized entity and agents is done through message exchanges allowing a bi-directional interaction between them. In one direction, the COE agent sends measured performance indicators and e2NB status to the centralized COE and other agents, while in the other direction the centralized entity enforces policies that define the operation to be executed by the agents and their underlying eNB and vUEs.

The above design provides the required flexibility in realizing the COE, and is able to reduce the control overhead by delegating more functions to the COE agent at the cost of less coordinated operation.

4.1 Topology Control

As we are using omni directional antennas, activating a link from a base station means interfering with all other links in all directions, regardless of the activated link. In order to remedy the interference from other links, we apply the power control and adaptive modulation and coding (AMC) scheme in *Un* interface that is already applied in *Uu* interface. To enable these two schemes, the received signal power and link quality shall be measured in *Un* interface.

Moreover, LTE allows for multi point-to-point transmissions from a single BS using OFDMA to transport unicast data to several nodes in the same slot instead of communication to each other nodes in per-slot basis. Via utilizing this property, the scheduler at each BS is only responsible to (a) select the neighbor BS(s) to transmit to and (b) pick the applied modulation and coding scheme (MCS) rather than applying topology-dependent policy that requires a priori topology information.

4.2 Global routing

We apply the Dijkstra's algorithm at COE to find the shortest path to route traffic in backhaul. Such algorithm could minimize per-flow latency provided that the traffic load is below the maximum throughput and avoid congestion. Nevertheless, a better performance can be achieved if an adaptive distributed mesh routing is used to cope with different traffic pattern and network topology [10].

4.3 Subframe scheduling

The main scheduling problem is to share the frequency (RBs) and time resources (MBSFN SFs) between the e2NBs. Such scheduling is achieved at two different levels in different time-scales. Algorithm 1 provides a high-layer scheduling approach that summarizes the two different levels. A centralized node scheduler (NS) (Algorithm 2) is executed periodically or on reaction to an event (for instance,

Algorithm 1: Coordinated Scheduling Algorithm

Input : Let P_{SF} be the SuperFrame update periodicity
Let $e2NB\_List$ be the list of e2NBs

```
begin
  while networkrunning do
    SF = SF + 1 /* current subframe identifier */
    TryNew = 0
    if event then
      TryNew = 1
    if SF ≡ 0 (mod P_SF) or TryNew == 1 then
      get_metrics()
      compute(L_SF) (c.f. Eq. 1)
      NeedMatched = 0
      while NeedMatched == 0 do
        compute(SF_realtime) (c.f. Eq. 2)
        compute(SF_elastic) (c.f. Eq. 3)
        [SF_TX, NeedMatched] =
          centralized_NS(SF_realtime, SF_elastic) (c.f. Alg. 2)
        L_SF = L_SF + 10
        if L_SF > MaxLat and TryNew == 1 then
          reject(flow) /*too many realtime flows*/
          compute(L_SF) (c.f. Eq. 1)
          NeedMatched = 0
    foreach u ∈ e2NB_List do
      local_LS(SF, SF_TX[u]) (c.f. Alg. 3)
```

a new flow with real-time requirements) and it defines on which MBSFN SFs each e2NB can transmit. Then, each e2NB integrates a local link scheduler (LS) that executes at every SF and takes care of the per-link scheduling (Algorithm 3).

4.3.1 Centralized node scheduling

Without loss of generality, we only consider a time-domain resource allocation (i.e., SF allocation) in DL direction (i.e., from eNB to vUE) among the neighboring nodes for self-backhauling, as UL SFs allocations are directly associated to allocated DL SFs. The proposed node scheduling determines for each BS, the active SFs within a superframe (defined as a multiple of a LTE frame) for which the transmission or reception are scheduled in accompany with the activated vUE(s). Its goal is to allocate enough SFs to each e2NB to fulfill the real-time traffic transportation. If that is not possible, Algorithm 1 guarantees that each e2NB gets at least one SF per superframe through iteration over the superframe length.

The resource allocation within a superframe for each e2NB is repeated until the node scheduling gets updated, which is triggered by traffic load and/or channel variabilities in the superframe. The length of the superframe (in milliseconds) is determined in such a way to satisfy the latency requirement for real-time traffic flows, and remains the same for all e2NBs in the network (generally in the order of tens of millisecond). It is computed as a function of expected maximum number of hops, denoted as M_{hops} in Eq. 1, that a given flow may experience.

$$L_{SF} = \lceil MaxLat/((M_{hops} + offset)) \rceil \tag{1}$$

The $MaxLat$ represents the maximum acceptable latency for the real-time flows (e.g. 150ms for VoIP). The $offset$ is the stretch factor of M_{hops}, and it depends mainly on the mobility pattern (i.e. speed) and network size.

It has to be mentioned that the algorithm relies on a priori knowledge of traffic flow (real-time or statistics) in the network to proportionally allocate resources among the base stations while respecting their QoS requirements. Some real-time flow control policy is required to avoid accepting too many flows that could overload the network. The following inputs are considered for each link:

- Expected traffic load of real-time flows;
- Expected link quality or the resulted MCS for each flow;
- Utilization ratio of allocated resources during the last node scheduling period.

A first metric, denoted as $SF_{realtime}[u]$, is computed to determine the number of SFs during a superframe required by a e2NB u to transmit the real-time flows. It is the main allocation control variable and is also used to sort the e2NBs in descending order before allocating the SFs. It is computed centrally as follows:

$$SF_{realtime}[u] = \frac{\sum_{v \in N(u)} PRB_{u,v}(tpt_{u,v} * L_{SF})}{N_{PRB}} \tag{2}$$

where $tpt_{u,v}$ is the sum of the throughput for real-time flows passing through link $u \rightarrow v$ (i.e., the transportation from node u to node v). $PRB_{u,v}(x)$ returns the number of PRBs required to transmit x bits given the current channel quality on link $u \rightarrow v$, $N(u)$ is a set that comprises all neighboring nodes of u and N_{PRB} is the total number of PRBs per SF.

A second metric is computed to increase the total number of allocated SFs per e2NBs, denoted as $SF_{elastic}[u]$, for links that are saturated due to elastic flows passing through resulting in a higher utilization ratio. We consider a link to be saturated if its ratio of saturated SFs (SFs that can transport less bits than what is queued) over allocated SFs is higher than a specified value, for instance 80%. $SF_{elastic}[u]$ is a simple estimator that tries to catch the relative needs of extra SFs among e2NBs. Eq. 3 details how it is obtained for a node u.

$$SF_{elastic}[u] = \left\lceil \frac{U[u]}{N_{freeSF} * \sum_u U[u]} \right\rceil \tag{3a}$$

$$U[u] = \sum_{v \in N_{sat}(u)} TBS_{PRB}(u, v) \tag{3b}$$

$$N_{freeSF} = \frac{L_{SF}}{10 * SF_{MBSFN}} - \frac{\sum_u SF_{realtime}[u]}{N_b / N_u} \tag{3c}$$

$U[u]$ represents the sum of expected transport block per PRB of all outgoing links from node u, where $N_{sat}(u)$ comprises the neighboring nodes of u with saturated link along the direction from u, $TBS_{PRB}(u, v)$ is the expected transport block size (TBS) per allocated PRB over the link $u \rightarrow v$, N_b is the total number of e2NB while N_u is the average number of direct neighbors of each e2NB during a superframe. N_{freeSF} represents the potential number of free SFs after allocating $SF_{realtime}$.

Algorithm 2 is used to allocate the SFs between the e2NBs. $SF\_TX[u][v]$ is the list of SF on which e2NB u can transmit to e2NB v (i.e, v will be in reception mode). $NeedMatched$ indicates if the scheduler was able to give at least $SF_{required}[u]$ SF(s) to e2NB u for all e2NBs. $N_{interference}(u)$ represents the interference area around u and can be adjusted from the adjacent neighbors to nodes within the radio coverage depending on the scheduling policy (e.g. conservative for a reliable transmission). In either case, the link adaption shall cope with the induced interference at the local scheduler.

4.3.2 Local scheduling

The local scheduler is priority-based dynamic round robin algorithm on per SF basis (see Algorithm 3), which takes into accounts the buffer status and priorities between flows as well as the channel quality. It first allocates resources for the backhaul links (i.e. vUEs) followed by the access links (i.e. UEs) for real-time flows. The remaining resources will be allocated to the elastic flows until all the buffers become empty or there is no PRB left.

4.4 Local Routing

With the proposed algorithm, we aim at providing a DL SF at each node. It means every node can reach all its connected nodes using DL, and with the associated UL SF, can be reached by these nodes using UL. As it applies to all nodes, every connected adjacent nodes have UL and DL paths to each other. It means that each packet can go over the DL or the UL path to reach the next e2NB. We use the expected waiting time of both UL and DL queues as the metric to decide at each packet arrival to which queue (DL or UL) to add it.

4.5 Interference measurement and link adaptation

The induced interferences are multi-facet: between neighboring e2NBs happening during the backhaul SFs, between e2NBs at UE side in DL and between UEs at e2NB side in UL during the access SFs. They are caused mainly because the neighboring e2NBs are not aware of the resource allocation policy of each other. There are two ways to deal with it. First, interference can be handled locally by each e2NB, simply by using AMC to adapt the MCS of transmissions based on the reported CQIs (for DL) and its own measurement (for UL) as is done by legacy systems. However, it can be improved for the backhaul by taking into account the fact that interference power will most probably be at the same level during repeating SFs in each superframe as the other scheduled nodes will be the same.

Second, it can be handled through cooperation between nodes to achieve a better performance by dynamically enforcing the resource allocation policy based on the eNB status information. This is in contrast to (semi-)statically resource allocation policy with no or little interactions among eNBs (as in case of X2 signaling to support ICIC [11]). Through self-backhauling capability, enabled by vUE through *Un* interface, eNBs can share the status information with the COE and neighboring eNBs including:

- the DL channel quality indicators (CQI), frequency and time resources;
- the UL interference status for each frequency resource (RB) experienced by the neighboring eNBs.

The above information allows to adapt the modulation and coding scheme (MCS), frequency resources, and the transmit power for both legacy UEs and vUEs so as to coordinate the resources and transmissions across access and backhaul links. For instance, an e2NB A is using high Tx power on frequency resources (f1) to transmit to e2NB B. With COE, e2NB C allocates a different frequency resources (f2) to transmit to e2NB D which suffers less interference from e2NB A.

5 Evaluation of the proposed approach on a LTE mesh network

This section provides the performance evaluation of the proposed coordinated scheduling and compares the results to a classical mesh network link scheduling algorithm [12].

5.1 Simulation environment

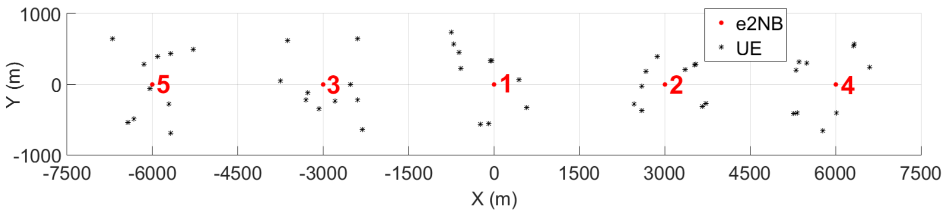
A complete LTE simulator is developed in Matlab allowing to create a 2D-map of an arbitrary network of e2NBs with their associated UEs and to generate arbitrary flows between every type of nodes (e.g., U2U, e2U, e2e traffic). We assume that it takes 5ms for a eNB to process an incoming RF packet before pushing it to the queue of the next interface according to its local routing.

5.1.1 Channel models

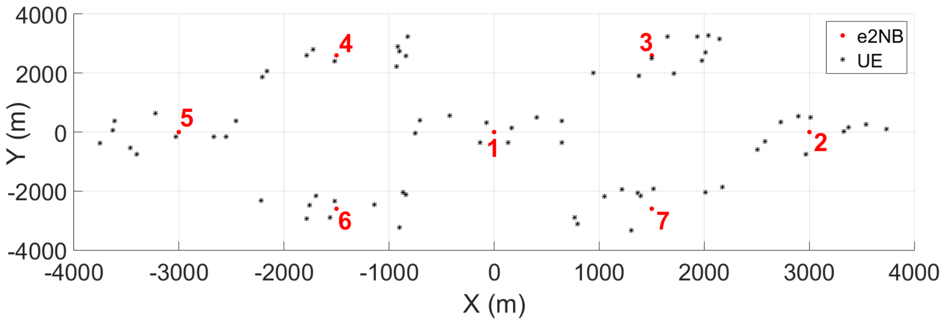
Between e2NBs, a freespace path loss model of coefficient 2.1 is applied with Claussen shadow fading and EPA channel type. Between e2NBs and UEs, a rural (TR 36.942) path loss model is used with Claussen shadow fading and EPA channel type. No further assumption is made on interference coordination and mitigation for UEs. e2NBs and UEs are assumed to be fixed over time and equipped with omni directional antennas. All interferences caused by concurrent transmission are taken into account at each receiver.

5.1.2 Topologies

We define two different topologies as shown in Fig. 3(a) and 3(b). In both topologies, each BS has 10 attached UEs and is connected to the closest adjacent e2NBs.



(a) Line Topology with 5 e2NBs



(a) Hexagonal Topology with 7 e2NBs

Figure 3: Considered network topologies

5.1.3 Traffic patterns

Two traffic patterns are considered. In the first, all the UEs are randomly paired with each other and establish a real-time VoIP call with a packet size of 20 bytes (40 bytes on PHY using RoHC) with an arrival rate of 20ms in both directions. In the second, two fixed BS to BS elastic data transfers are added to the first traffic pattern.

Each elastic traffic tries to maximize its bandwidth and behaves as a connection-oriented acknowledgment service in that the new packet will be only generated if the maximum number of non-acknowledged packets is not reached. This type of flow represents inter-sites (i.e. BSs) data transfers that often happen in military and public safety application scenarios and are not directly generated by UEs to avoid limited UL rates.

5.1.4 Other settings and parameters

HARQ mechanism is not implemented in the simulator. UEs are only scheduled in UL/DL transmission in non MBSFN SFs. Expired VoIP packets (one-way-delay larger than 200ms) are dropped from the queue to improve the performance. A link is considered to be saturated if the utilization ratio is larger than 80%. Each e2NB is configured for 10MHz channel bandwidth in band 4 (2.1GHz). For the BSs, the minimum transmit power is set to 20dBm (for Li algorithm) and the maximum to 40dBm for DL UE transmissions and 46dBm for inter-e2NB transmissions. For UEs, maximum UL transmit power is 23dBm.

5.1.5 Li and Ephremides algorithm

In [12], Li and Ephremides proposed a joint scheduling, power control and routing algorithm for TDMA wireless mesh networks. Their approach relies on a centralized scheduler that runs at each time slot and decides which links should be activated as well as their transmitted powers. Their proposed algorithm first sorts all possible links according to a link metric, which is based on the queue size of itself and all other links blocked by this link to prioritize the large queues while handling the congestion. Here, a link (a) is considered to be blocked by a link (b) if its source or destination is identical to either the source or destination of link (b). Then, an iterative algorithm finds the set of links with the highest priorities that can be concurrently activated while respecting their power limit and SINR requirements given all induced interference. In addition, their method periodically updates the routing paths based on the link quality and the relative queue size of each link to allow packets to be redirected towards less congested links when queue buffer occupancy increases.

One of the main drawbacks of Li and Ephremides approach is that the link scheduling is done in per-slot basis and thus it requires a perfect knowledge of queue sizes of all nodes and channel behavior between all nodes. Moreover, each activated node is used for a single point-to-point (P2P) link which might not be efficient in LTE as multiple P2P transmissions are allowed in both DL and UL directions at the same slot.

5.1.6 Improvement in Li and Ephremides algorithm

The following improvement is implemented to apply the Li and Ephremides algorithm in LTE. We update the link metric and take into account both DL and

UL queues of each link. Inter-node link scheduling is done for both DL and UL. The iterative power allocation algorithm is modified to incorporate the adaptive modulation and coding of different SINR regimes (instead of minimum SINR).

5.2 Algorithms considered

Three global approaches are considered for the comparison: (1) the Li and Ephremides algorithm with the above modifications (baseline), (2) Algorithm 2 used to equally allocate MBSFN SFs between nodes with fixed number of SF requirement in both real-time ($SF_{realtime}$) and elastic ($SF_{elastic}$) traffic among all e2NBs (denoted as the Basic algorithm), and (3) the proposed Algorithm 2 as described in section 4. In terms of local scheduler, (1) uses Algorithm 3 for UE scheduling but allocates all RBs of inter-e2NB links to one vUE at a time while (2) and (3) use Algorithm 3 to allocate RBs in a fined-grained manner. In all cases, the incoming packets are classified either as real-time VoIP or elastic data traffics and can be sorted in the local vUE queues based on two manners: (a) the number of hops to reach the destination, labeled as *Hops* in the figures, or (b) the shortest deadline, labeled as *AirTime*. This affects the number of dropped VoIP packets, and consequently the throughput of the elastic traffic.

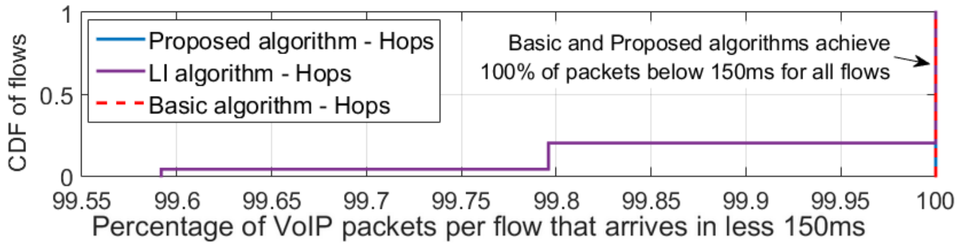
5.3 Results

Firstly, we consider two scenarios: (a) only VoIP traffic and (b) both VoIP and elastic traffics. Then, we show the results in two forms: (i) the CDF plot of the percentage of VoIP packets that are within the pre-defined maximum delay (150ms) and (ii) throughput of elastic traffic flows.

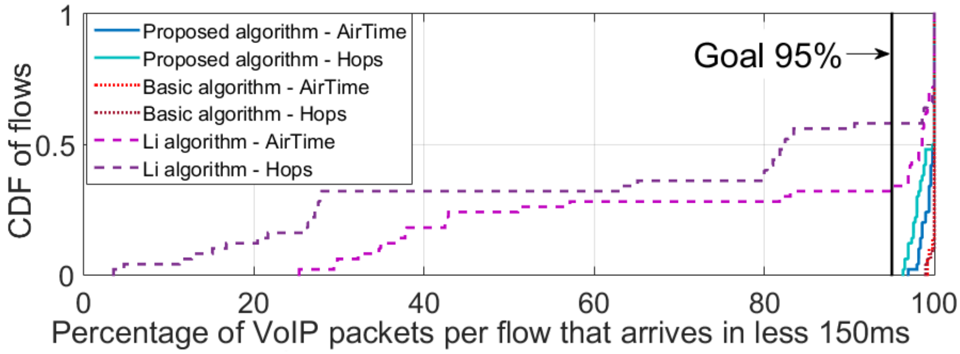
5.3.1 Line topology

From Fig. 4.(a), it can be seen that with only VoIP traffic, all approaches satisfy the required 95% of packet with less than 150ms end-to-end latency on all the VoIP flows. This is because there is sufficient capacity to transport all flows even without using the multi point-to-point capabilities of LTE, particularly because the topology allows for a maximum of two point-to-point communications (two adjacent neighbors) from a given node at a time slot.

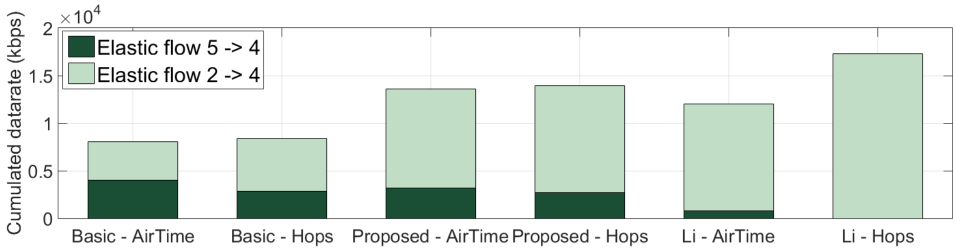
From Fig. 4.(b), we can see that Li approach performs poorly regarding latency requirements as only 60% of flows meet the requirement when using *AirTime* as a sorting metric and only 40% of flows meet the requirement when using *Hops* as a sorting metric. On the other hand, both *Basic* (2) and *Proposed* (3) approaches allows all flows to meet their latency requirement, with *Basic* performing a little better than the *Proposed* approach and *AirTime* being the sorting metric performing the best for both.



(a) CDF of VoIP flows. (50 VoIP flows)



(b) CDF of VoIP flows. (50 VoIP plus 2 inter-e2NB elastic flows)



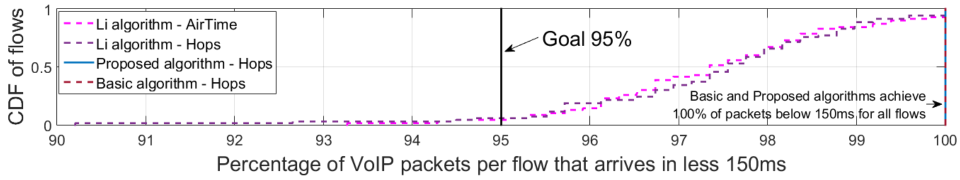
(c) Cumulated data rate of elastic traffic with different enforced policy. (50 VoIP plus 2 inter-e2NB elastic flows)

Figure 4: Line Topology

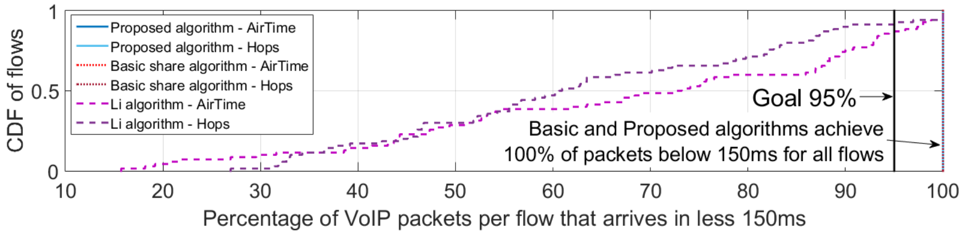
From Fig. 4.(c), it can be observed that the *Basic* approach has the lowest cumulated throughput as its fixed global allocation does not react to elastic flows, while the *Proposed* approach falls short behind Li and Ephremides approach.

5.3.2 Hexagonal topology

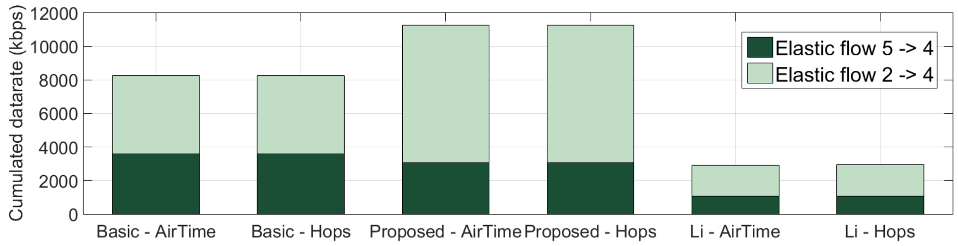
For the hexagonal topology, we observe in the Fig. 5.(a) that around 5% of VoIP flows do not meet their latency requirement when using Li approach while all flows under *Basic* and *Proposed* approaches respect it. In the Fig. 5.(b), again approaches (2) and (3) performs perfectly regarding VoIP requirements but Li approach performs poorly with almost 90% of the VoIP flows failing to meet the requirements. From Fig. 5.(c), we observe that it is Li approach that performs the worst while the *Proposed* approach achieves 35% throughput improvement over the *Basic* approach. The under-performance of Li approach is due to its lack of multi P2P capability that is highly beneficial in small network with central position of one e2NB.



(a) CDF of VoIP flows. (70 VoIP flows)



(b) CDF of VoIP flows. (70 VoIP plus 2 inter-e2NB elastic flows)



(c) Cumulated data rate of elastic traffic with different different enforced policy. (70 VoIP plus 2 inter-e2NB elastic flows)

Figure 5: Hexagonal topology

It can be observed that for both topologies, the *Proposed* approach (3) achieves the best latency-throughput trade-off for real-time and elastic traffics thanks to it capability to adapt to link requirements and to the use of multi P2P link scheduling.

6 Conclusion

In this article, we present a method to efficiently and practically realize an in-band LTE wireless mesh networks that leverages the LTE relay interface (*Un*) and takes into account QoS requirements of real-time flows. We then compare that approach to two other ones in simple but yet realistic scenarios. The findings are dual: first, we show that in-band meshing of LTE base stations is possible without requiring intense modifications on current standards while keeping legacy UE support. Second, we show that our approach is able to guarantee QoS requirements of real-time flows while improving the global network throughput for elastic flows. In future, we plan to realize radio-frequency experiments to assess the performance of the approach.

References

[1] R. Favraud, A. Apostolaras, N. Nikaein, and T. Korakis, “Toward moving public safety networks,” *IEEE Communications Magazine*, vol. 54, no. 3, pp. 14–20, March 2016.

[2] R. Favraud and N. Nikaein, “Wireless mesh backhauling for lte/lte-a networks,” in *IEEE MILCOM 2015*, Oct 2015, pp. 695–700.

[3] R. Favraud and N. Nikaein, “Analysis of LTE Relay Channels for Self-Backhauling in LTE Mesh Networks,” Eurecom, Tech. Rep. RR-17-331, April 2017.

[4] Y. Yuan, *LTE-Advanced Relay Technology and Standardization*. Springer-Verlag Berlin Heidelberg, 2013.

[5] A. S. Mohamed, M. Abd-Elnaby, and S. A. El-Dolil, “Self-organized dynamic ffr resource allocation scheme for lte-advanced relay based networks,” *Wireless Personal Communications*, vol. 91, no. 2, pp. 933–955, 2016.

[6] J. Han and H. Wang, “Uplink performance evaluation of wireless self-backhauling relay in lte-advanced,” in *WiCOM 2010*, Sept 2010.

[7] P. H. Pathak and R. Dutta, “A survey of network design problems and joint design approaches in wireless mesh networks,” *IEEE Communications Surveys Tutorials*, vol. 13, no. 3, pp. 396–428, Third 2011.

[8] A. F. Ribadeneira, “An analysis of the MOS under conditions of delay, jitter and packet loss and an analysis of the impact of introducing piggybacking and reed solomon FEC for VoIP,” Master’s thesis, Georgia State University, 2007.

[9] X. Foukas, N. Nikaein, M. M. Kassem, M. K. Marina, and K. Kontovasilis, “Flexran: A flexible and programmable platform for software-defined radio access networks,” in *ACM CoNEXT 2016*, 2016.

[10] I. F. Akyildiz, X. Wang, and W. Wang, “Wireless mesh networks: A survey,” *Comput. Netw. ISDN Syst.*, vol. 47, no. 4, pp. 445–487, Mar. 2005.

[11] *X2 General Aspects and Principles (Release 19)*, 3GPP Technical Specification 36.420, 2009.

[12] Y. Li and A. Ephremides, “A joint scheduling, power control, and routing algorithm for ad hoc wireless networks,” *Ad Hoc Networks*, vol. 5, no. 7, pp. 959 – 973, 2007.

Algorithm 2: Centralized Node Scheduler (centralized_NS)

Input : Let $MBSFN$ be the list of MBSFN SFs during the scheduling period L_{SF}.
Let $e2NB\_List$ be the list of e2NBs in the network.
Let $N[u]$ be the list of neighboring e2NBs of e2NB u.
Let $LINK(u, v)$ be a link between two e2NBs where e2NB u acts as an eNB and e2NB v as a vUE.
Let $SF\_RX[u]$ be the list of available RX SFs at u.
Let $SF\_TX[u][v]$ the list of available TX SFs at u for each connected e2NB v.

Output: SF_TX,$NeedMatched$

```
foreach u ∈ e2NB_List do
    SF_RX[u] = MBSFN. ∀v, ∀SF, SF ∈ SF_TX[u][v].
    haveOne[u] = 0. realtime[u] = SF_realtime[u].
    elastic[u] = SF_elastic[u]
NeedMatched = 0
foreach SF ∈ MBSFN do
    sort_descending(e2NB_List, realtime)
    sort_descending(u ∈ e2NB_List | realtime[u] == 0, elastic)
    foreach u ∈ e2NB_List do
        if realtime[u] > 1 or elastic[u] > 1 then
            if SF ∈ SF_TX[u][*] then
                Transmit = 0
                foreach v such that LINK(u, v) exist do
                    if SF ∈ SF_RX[v] then
                        Transmit = 1
                        remove(SF, SF_RX[u])
                        remove(SF, SF_TX[v][*])
                        foreach w ∈ N_interference[u] do
                            remove(SF, SF_RX[w])
                        foreach w ∈ N_interference[v] do
                            remove(SF, SF_TX[w][*])
                    else
                        remove(SF, SF_TX[u][v])
                if Transmit == 1 then
                    haveOne[u] = 1
                    if realtime[u] > 1 then
                        realtime[u] − −
                    else
                        elastic[u] − −
    if ∀u realtime[u] == 0 then
        NeedMatched = 1
    if ∀u elastic[u] == 0 then
        foreach u ∈ e2NB_List do
            elastic[u] = 1
```

Algorithm 3: Local Link scheduler (local_LS)

Input : Let u be the current e2NB identifier.
Let SF be the current subframe identifier.
Let UE_List and vUE_List be the list of UEs/vUEs with non empty queues at u.
Let $Q[x][p]$ be the queue size of UE/vUE x for flows of priority p.
Let N_{PRB} be the number of available PRBs in SF.
Let $SF\_TX[u][x]$ be the list of available TX SFs at u for each $x \in UE\_List \cup vUE\_List$.

Output: PRBs allocated for UEs and vUEs of e2NB u

Result: u uses $PRB[x]$ to transmit to each $x \in UE\_List \cup vUE\_List$

$sort\_descending(UE\_List, Q[*][0])$
$sort\_descending(vUE\_List, Q[*][0])$
$List = append(vUE\_List, UE\_List)$
$prio = 0$ /* 0: real-time, 1: elastic */
while $N_{PRB} > 0$ *and* $prio < 2$ **do**
    $increase\_prio = 1$
    foreach $x \in List$ **do**
        if $SF \in SF\_TX[u][x]$ **then**
            if $N_{PRB} > 0$ *and* $TBS[x] < \sum_{b=0}^{prio} Q[x][b]$ **then**
                $PRB[x] + +$
                $N_{PRB} - -$
                $TBS[x] = TBS(mcs[x], PRB[x])$
                $increase\_prio = 0$
    if $increase\_prio == 1$ **then**
        $prio + +$