Efficient Management of Interference and Power by Jointly Configuring ABS and DRX in LTE-A HetNets

You-Chiun Wang and Chien-Chun Huang

Abstract—Long term evolution-advanced (LTE-A) is a mobile communication standard widely used for broadband wireless access. It allows large macrocells and small picocells coexisting in a *heterogeneous network (HetNet)* to increase throughput and facilitate deployment. To handle signal interference between cells, LTE-A enhances channel quality of picocells by suspending the macrocell's transmission in an *almost blank subframe (ABS)*. Moreover, LTE-A adopts the *discontinuous reception (DRX)* method to let a user equipment (UE) adaptively disable its transceiver to save energy. However, existing studies view ABS and DRX as independent methods and address cell interference and energy saving separately. The paper points out that they have much in common, and develops a *joint interference and power management (JIPM) mechanism* to improve LTE-A performance by integrating ABS and DRX. Based on channel conditions and traffic demands of UEs, JIPM computes the amount of resources given to each UE and adjusts the parameters of ABS and DRX accordingly. Simulation results show that JIPM achieves higher energy efficiency and lower packet dropping of real-time flows, as compared with other schemes.

Index Terms—ABS, DRX, LTE-A.

1 Introduction

Thanks to the popularity of mobile phones and wireless networks, people desire to freely access the Internet today. Most operators thus provide *long term evolution-advanced (LTE-A)* services for high-speed communications. To guarantee seamless coverage while serving many *user equipments (UEs)* in hotspots, operators deploy picocell base stations (also called *eNBs*) within macrocells, and make them cooperate to form a *heterogeneous network (HetNet)*. HetNet has many advantages such as flexible deployment [1] and load sharing [2]. It is also a trend to deploy numerous picocells to strengthen HetNets for the upcoming 5G epoch [3].

However, signal interference between a macrocell and its picocells is a big problem in HetNets, as the macrocell eNB emits much larger transmitted power than a picocell eNB. To conquer the problem, LTE-A uses the *enhanced inter-cell interference coordination (eICIC)* technique consisting of frequency-based and time-based methods. The frequency-based method asks macrocell and picocell eNBs to send data via different bands for interference avoidance. The time-based method aims to reduce the macrocell's interference exerting on picocell UEs. Specifically, the macrocell eNB picks a number of subframes[1] as *almost blank subframes (ABSs)* in a cycle, where it sends nothing but low-powered control signals. In this way, picocell UEs can receive data from their eNBs with very little interference from the macrocell eNB. Owing to its better utilization of the spectrum resource, the time-based method is more flexible and popular than the frequency-based one [4], [5], and it is also known as the *ABS method*.

How to reduce energy consumption of UEs is also critical. LTE-A deals with this issue by the *discontinuous reception (DRX)* method, which makes a UE alternate between *sleeping* and *awake* states. In the sleeping state, the UE turns off its transceiver to save energy (and it stops listening to the channel accordingly). In case that there are downlink data for the UE, the eNB keeps the data until the next time that the UE wakes up. Once the UE does not receive any data after passing a predefined number of such cycles, it can prolong the cycle's length with a longer duration of the sleeping state to conserve more energy. When the UE gets data from the eNB (in the awake state), the cycle's length is reset again. The DRX parameters such as cycle length and awake time are controlled by the eNB via sending management packets to the corresponding UEs. Thus, the eNB is aware of the current DRX state of each UE in its cell [6].

On the face of it, ABS and DRX seem to be two independent methods for interference and power management in LTE-A, respectively. In fact, they do share some similarities in terms of UEs' behavior. Specifically, macrocell UEs cannot get data in an ABS, while a UE disables its transceiver in the sleeping state. However, existing studies aim to improve either the ABS or DRX methods individually. They may make macrocell UEs keep awake in ABSs, which wastes their energy. Even worse, picocell UEs may go to sleep when they have good channel quality (as the macrocell eNB stops data transmissions), which wastes the spectrum resource. Obviously, if the parameters of ABS and DRX can be co-adjusted properly such that UEs sleep and wake up at the *right* time (depending on their channel conditions), we can make a good balance between network throughput and energy consumption of UEs.

Therefore, this paper proposes a *joint interference and power management (JIPM) mechanism* to improve performance of LTE-A HetNets by integrating ABS and DRX through co-deciding their parameters. JIPM first synchronizes both ABS and DRX cycles. Based on the channel quality and traffic demand of each UE, it estimates the amount of resources required by

The authors are with the Department of Computer Science and Engineering, National Sun Yat-sen University, Kaohsiung, 80424, Taiwan. Email: ycwang@cse.nsysu.edu.tw; m043040008@student.nsysu.edu.tw.

1. A subframe is the time unit defined in LTE-A, whose length is 0.5 ms.

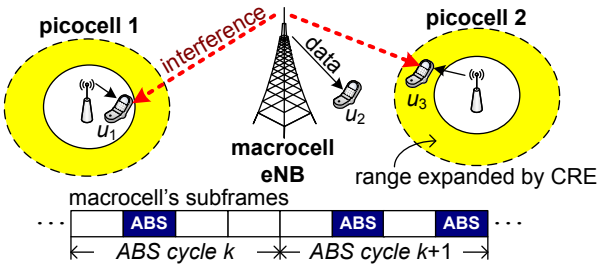


Fig. 1: Interference management by the ABS method.

the UE. Then, JIPM computes a suitable number of ABSs to balance macrocell and picocell loads, and decides the DRX parameters accordingly. Through simulations, we show that the JIPM mechanism not only saves more energy of UEs but also increases network throughput, thereby greatly improving energy efficiency. Besides, JIPM can alleviate packet dropping of real-time flows as compared with other methods. Our contributions are to point out the benefits of integrating ABS and DRX and also develop an efficient mechanism to significantly improve performance of LTE-A HetNets.

This paper is outlined as follows: Section 2 briefly introduces LTE-A and Section 3 surveys related work. We propose the JIPM mechanism in Section 4. Then, Section 5 evaluates its performance. Finally, Section 6 concludes the paper and discusses future work.

2 LTE-A Overview

2.1 Resource Management

LTE-A divides the spectrum resource into a 2D array of *physical resource blocks (PRBs)* for management, where each PRB has 0.5 ms duration and 180 kHz bandwidth. PRBs are exclusive when SISO or SU-MIMO techniques are used[2], and no two UEs can share the same PRB. The eNB takes charge of allocating PRBs to UEs in each *transmission time interval (TTI)*, whose length is 1 ms. If the channel has bandwidth of 1.4, 3, 5, 10, 15, or 20 MHz, the eNB offers 6, 15, 25, 50, 75, or 100 PRBs in a TTI, respectively. The amount of data sent by a PRB depends on its *modulation and coding scheme (MCS)*. A more complex MCS lets the PRB send more data, but it needs less interference, and vice versa. Each UE reports the *channel quality indicator (CQI)* that reveals its channel condition to help the eNB choose MCS.

A UE may have multiple flows, each with its own QoS (quality of service) demand. LTE-A uses *QoS class identifier (QCI)* to describe two QoS parameters for a flow: delay budget and loss rate, which give the longest delay time that the flow can tolerate and the maximum allowable probability that its packets are dropped, respectively. LTE-A classifies flows into *guaranteed-bit-rate (GBR)* and *non-GBR* ones. GBR flows often support real-time services with strict delay demands (e.g., VoIP and video), while non-GBR flows are used for applications with loose deadlines (e.g., web browsing). Thus, GBR flows would have smaller QCIs and delay budgets than non-GBR ones.

2.2 Interference Management

A goal of HetNet is to share loads of macrocell eNBs by deploying picocells in their communication ranges. However, the transmitted power of a macrocell eNB is typically 46 dBm while that of a picocell eNB is at most 30 dBm [7]. Thus, UEs prefer connecting to macrocell eNBs due to their high *reference signal received power (RSRP)*, which conflicts with the goal. To conquer this problem, LTE-A uses *cell range expansion (CRE)* by adding a small bias ψ to the RSRP values of picocells. Specifically, let $\mathcal{E}$ be the set of eNBs. A UE u_i selects an eNB e_j to connect based on

$$e_j = \arg\max_{e_j \in \mathcal{E}} (\xi_{i,j} + \{\psi \,|\, \text{if } e_j \text{ is a picocell eNB}\}), \qquad (1)$$

where $\xi_{i,j}$ is u_i's RSRP from e_j. Fig. 1 gives an example, where picocell 2 *virtually* expands its range by CRE, so it can serve u_3 and share the macrocell's traffic load.

However, the macrocell eNB imposes large signal interference on picocell UEs, especially the UEs in the CRE area. Thus, LTE-A proposes the ABS method to reduce such interference. It divides the time axis into *ABS cycles*, during which some subframes are selected as ABSs. In an ABS, the macrocell eNB sends only low-powered control signals. Thus, picocell UEs incur little interference from it and can use complex MCSs to get data in a high rate. Fig. 1 shows the ABS method, where u_1 and u_3 improve their channel quality but u_2 gets no data in ABSs. The *ABS ratio* (denoted by δ) decides how many ABSs are selected in a cycle. As macrocell UEs have no throughput in ABSs, this ratio will affect network performance and fairness. LTE-A allows a macrocell eNB dynamically adjusting its δ value to facilitate interference management.

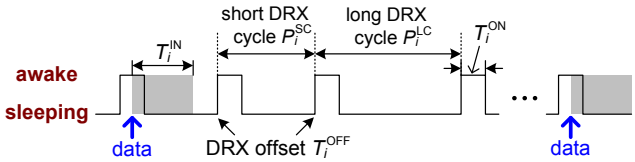


Fig. 2: Power management by the DRX method.

2.3 Power Management

LTE-A uses DRX for power management, which cuts the time axis into DRX cycles, as shown in Fig. 2. In each cycle, a UE listens to the *physical downlink control channel (PDCCH)* for a while to check if there are data sent from the eNB. If not, the UE turns off its transceiver to save energy until the next cycle. In this case, the eNB will keep the UE's data if necessary. There are two types of DRX cycles: *short* and *long*. If a UE does not receive any data after some short cycles, it changes to the long cycle to extend its sleeping time to conserve more energy.

In Fig. 2, the awake and sleeping time of a UE u_i is decided by six parameters: 1) *DRX inactivity timer* (T_i^{IN}) indicates how many subframes that u_i should stay awake when it receives data from PDCCH, 2) *short DRX cycle* (P_i^{SC}) is the length of a short cycle, 3) *DRX short cycle timer* (T_i^{SC}) gives the deadline for u_i to change to a long cycle if it keeps receiving nothing from PDCCH, 4) *long DRX cycle* (P_i^{LC}) is the length of a long cycle, and 5) *on-duration timer* (T_i^{ON}) gives the number of subframes that u_i should stay awake in the beginning of a DRX cycle, and 6) *DRX offset* (T_i^{OFF}) points out when a cycle begins. By sending *radio resources configuration (RRC)* messages to a UE which include the DRX parameters, the eNB can control the sleeping behavior and energy consumption of that UE.

2. SISO: single-input single-output, SU-MIMO: single-user multiple-input multiple-output

3 Related Work

3.1 ABS Configuration

Several studies find the optimal ABS ratio δ and CRE bias ψ with a *full-buffer* model, where each queue always has packets to be sent. The work [8] finds two sets of UEs to get data in ABSs and non-ABSs by dynamic programming, and derives δ to keep UEs' fairness. Trabelsi et al. [9] search all possible (δ, ψ) pairs, and pick the pair to maximize throughput. The study [10] assumes that UEs arrive to the network via a Poisson process and they have only FTP flows. It uses the response surface method, which is a stochastic optimization algorithm, to compute δ and ψ. However, not every flow follows the full-buffer model (e.g., VoIP and videos).

How to adjust δ to react to a dynamic network situation is also addressed. The work [11] raises δ with a probability when there are more UEs in picocells, but it does not consider traffic demands of UEs. In [12], both throughput and fairness maximization problems are modeled by sum-rate and product-rate utilities, respectively, and the ABS ratio is decided by data rates and queue lengths of the UEs located in macrocells, picocells, and CRE ranges. Lu et al. [13] propose an interference index defined by the difference of throughput when UEs incur signal interference. They add or subtract δ by a constant depending on the index. A fuzzy logic system is adopted in [14] to compute the ABS ratio, whose inputs contain UE number, macrocell throughput, and picocell channel quality. The work [15] formulates the problem of ABS configuration by a general-form consensus problem, and uses ADMM (alternating direction method of multiplier) to find a solution. However, the issue of packet latency is not discussed in these studies.

A few studies configure ABS with the goal of reducing GBR packet delay. The work [16] uses a genetic algorithm to allocate ABSs, whose fitness function takes cell throughput, noise interference, and video latency as inputs. Wang et al. [17] find a network-status indicator to check the suitability of using ABSs based on the capacity of each cell, and analyze queued data with different urgent levels to set the ABS ratio, so as to alleviate GBR packet dropping. However, [16] and [17] do not integrate ABS with DRX to balance between energy consumption and throughput of UEs.

3.2 DRX Configuration

Some studies set DRX parameters based on traffic loads or channel variations. The work [18] extends T_i^{ON} to shorten sleeping time of UEs when their loads grow. Besides, if a UE's channel condition keeps good for a while, it reduces T_i^{IN} for energy saving. In [19], the UEs with larger CQIs are given with a smaller T_i^{IN} value to conserve energy, as they can fast get data with more complex MCSs. However, both studies do not consider QoS demands of UEs.

How to configure DRX with delay concern is also discussed. The work [20] estimates the average delay by $(P_i^{\mathrm{LC}} - T_i^{\mathrm{ON}})/2$ and the power-saving ratio by $(P_i^{\mathrm{LC}} - T_i^{\mathrm{ON}})/(P_i^{\mathrm{LC}} + T_i^{\mathrm{IN}})$. It proposes two scenarios: 1) power-saving maximization with delay constraint and 2) delay minimization with power-saving constraint, to decide the values of P_i^{LC}, T_i^{ON}, and T_i^{IN}. In [21], a discrete-time Markov process is adopted to model the Internet traffic. By finding state transition probabilities from the process, it decides P_i^{LC}, P_i^{SC}, and T_i^{SC} to reduce energy consumption of UEs and meet their delay demands. The study [22] adjusts T_i^{ON}, T_i^{IN}, and T_i^{OFF} by referring to packet delay and CQI of each UE. If a UE has small packet delay and large CQI, it reduces both T_i^{IN} and T_i^{ON} while raising T_i^{OFF} to save energy, as the UE can receive packets in a short time.

A number of studies combine DRX with resource scheduling. The work [23] sets DRX cycles (i.e., P_i^{LC} and P_i^{SC}) of a UE as integer multiples of others to reduce the awake time of UEs due to resource competition. Then, the eNB allocates PRBs to the UEs whose packets will be dropped or timer T_i^{IN} will expire in the next subframe, so as to catch packet deadlines. Tung et al. [24] set P_i^{SC} as the minimum delay budget of a UE and T_i^{ON} as the expected transmission time based on its CQI. Given the UE's priority from a resource scheduling method, [24] adds a small value to the priority if the UE will go to sleep in the next subframe and the network load exceeds a threshold. The study [25] schedules multicast groups and configures DRX parameters such that the delay constraint of each multicast stream is not violated and total wake-up subframes of UEs can reduce. To do so, [25] decides the allocation order of multicast groups, including cycle lengths and offsets, to meet their delay demands. Then, it tunes both T_i^{IN} and T_i^{ON} of each UE in a group to reduce the wake-up time. The work [26] aims to support QoE (quality of experience) of GBR traffics and save energy of UEs. It proposes an opportunistic method to allocate resources to UEs, which considers multiple parameters including QoE requirement, channel condition, average throughput, buffer length, DRX status, and GBR/non-GBR flows.

However, the above studies neither consider the HetNet scenario nor address signal interference when UEs are awake. It motivates us to develop the JIPM mechanism by integrating both ABS and DRX for joint management of interference and power in LTE-A HetNets.

4 The Proposed JIPM Mechanism

4.1 Network Model

In the ABS method, each macrocell decides its ratio δ. For ease of explanation, our discussion aims at a *basic* HetNet with some picocells enclosed by one macrocell, and the result can be easily extended to a general HetNet with multiple macrocells. Let us denote by $\mathcal{E}$ and $\mathcal{E}_p$ the sets of all and picocell eNBs in the basic HetNet, respectively. All eNBs operate on the same frequency band and allot non-sharable PRBs to UEs by using SU-MIMO. Besides, each UE associates with only one eNB, and we denote by $\mathcal{U}_j$ the set of UEs linking to eNB $e_j \in \mathcal{E}$. A UE may generate GBR or non-GBR flows, where GBR flows have stringent delay requirements. To save their energy, we consider using long DRX cycles for UEs.

Given traffic demands of UEs, our problem asks how to decide the ABS ratio δ for the HetNet and also DRX cycle length P_i^{LC}, on-duration timer T_i^{ON}, and DRX offset T_i^{OFF} of each UE such that energy efficiency is maximized while GBR packet dropping is minimized. Specifically, energy efficiency is defined by the ratio of HetNet throughput to energy consumption of UEs. Besides, a GBR packet will be dropped if it is overdue. Table 1 summarizes our notations.

4.2 Mechanism Design

The JIPM mechanism comprises four stages to configure ABS and DRX parameters:

1) **CQI evaluation:** We synchronize ABS and DRX cycles, and compute CQIs of UEs in ABSs and non-ABSs.

TABLE 1: Summary of notations.

notation	definition
$\mathcal{E}, \mathcal{E}_p$	the set of all/picocell eNBs in a basic HetNet
$\mathcal{U}_j$	the set of UEs associating with eNB e_j
$P^{\mathrm{ABS}}, P_i^{\mathrm{LC}}$	ABS cycle length and DRX long cycle length
$T_i^{\mathrm{ON}}, T_i^{\mathrm{OFF}}$	on-duration timer and DRX offset of UE u_i
$\Upsilon_{i,j}^{\mathbf{O}}, \Upsilon_{i,j}^{\mathbf{A}}$	SINR of UE u_i from eNB e_j in a non-ABS/ABS
$Q_i^{\mathbf{O}}, Q_i^{\mathbf{A}}$	CQI of UE u_i in a non-ABS/ABS
$F(b_i, Q_i)$	a function to find the number of PRBs to send b_i data bits with CQI Q_i
$n_i^{\mathbf{G},\mathbf{O}}, n_i^{\mathbf{G},\mathbf{A}}$	the number of PRBs to send UE u_i's GBR data in a cycle with non-ABSs/ABSs
$n_i^{\mathbf{NG},\mathbf{O}}, n_i^{\mathbf{NG},\mathbf{A}}$	the number of PRBs to send UE u_i's non-GBR data in a cycle with non-ABSs/ABSs
N_j	available PRBs provided by eNB e_j in a cycle
L_j	traffic load of eNB e_j for GBR or non-GBR flows
γ	the number of ABSs in a cycle (ABS ratio: $\delta = \gamma / P^{\mathrm{ABS}}$)

2) **Resource estimation:** With CQI and cycle length, we estimate how many PRBs are required to satisfy the traffic demand of each UE.
3) **ABS setting:** Based on the amount of traffics in cells, we find the ABS ratio δ to raise throughput and balance loads, and also allocate PRBs to UEs.
4) **DRX setting:** Both T_i^{ON} and T_i^{OFF} of UEs are then decided by their allocated resources.

4.2.1 CQI Evaluation

Let $D_i^{\min}$ be the minimum delay budget of all flows owned by a UE u_i. Then, both ABS cycle length P^{ABS} and DRX cycle length P_i^{LC} are set by

$$P^{\mathrm{ABS}} = P_i^{\mathrm{LC}} = \min_{u_i \in \mathcal{U}} \{D_i^{\min}\}, \text{ where } \mathcal{U} = \bigcup\nolimits_{e_j \in \mathcal{E}} \mathcal{U}_j. \quad (2)$$

In other words, the lengths of these cycles are decided by the most stringent delay demand of flows in the network, so as to catch their packet deadlines.

We then evaluate the *signal-to-interference-plus-noise ratio (SINR)* of each UE. Suppose that eNB e_j sends data to UE u_i over subchannel c_k. Thus, u_i's SINR in a non-ABS is

$$\Upsilon_{i,j,k}^{\mathbf{O}} = \frac{S(u_i, e_j, c_k)}{\varepsilon N_0 B_k + \sum_{e_h \in \mathcal{E}, e_h \neq e_j} S(u_i, e_h, c_k)}, \quad (3)$$

where $S(u_i, e_j, c_k)$ is the strength of power received by u_i from e_j over c_k, which depends on e_j's transmitted power and the distance between u_i and e_j. In Eq. (3), the calculation of SINR $\Upsilon_{i,j,k}^{\mathbf{O}}$ considers not only the interference $S(u_i, e_h, c_k)$ from other eNBs but also the effect of environmental noise $\varepsilon N_0 B_k$, where ε is a constant noise figure, N_0 is the power spectral density of noise (usually set to -174 dBm/Hz), and B_k is the bandwidth of subchannel c_k (i.e., 15 kHz). On the other hand, there are two cases to compute u_i's SINR in an ABS:

- u_i is a macrocell UE: Since the macrocell eNB does not send user data, its transmitted power can be ignored. By Eq. (3), the SINR $\Upsilon_{i,j,k}^{\mathbf{A}}$ will be very small.
- u_i is a picocell UE: Only the interference from other picocell eNBs should be considered. Thus, so its SINR is computed by

$$\Upsilon_{i,j,k}^{\mathbf{A}} = \frac{S(u_i, e_j, c_k)}{\varepsilon N_0 B_k + \sum_{e_h \in \mathcal{E}_p, e_h \neq e_j} S(u_i, e_h, c_k)}, \quad (4)$$

TABLE 2: MCS supported by each CQI and its required SINR.

CQI	MCS	SINR
1	QPSK (78/1024)	-6.936 dB
2	QPSK (120/1024)	-5.147 dB
3	QPSK (193/1024)	-3.180 dB
4	QPSK (308/1024)	-1.253 dB
5	QPSK (449/1024)	0.761 dB
6	QPSK (602/1024)	2.699 dB
7	16QAM (378/1024)	4.694 dB
8	16QAM (490/1024)	6.525 dB
9	16QAM (616/1024)	8.573 dB
10	64QAM (466/1024)	10.366 dB
11	64QAM (567/1024)	12.289 dB
12	64QAM (666/1024)	14.173 dB
13	64QAM (772/1024)	15.888 dB
14	64QAM (873/1024)	17.814 dB
15	64QAM (948/1024)	19.829 dB

Given u_i's SINR over each subchannel c_k, its overall SINR in a non-ABS can be estimated by the *exponential effective SINR mapping (EESM)* approach [27]:

$$\Upsilon_{i,j}^{\mathbf{O}} = -\beta \ln \frac{1}{\tilde{m}_{\mathrm{C}}} \sum\nolimits_{k=1}^{\tilde{m}_{\mathrm{C}}} \exp(-\Upsilon_{i,j,k}^{\mathbf{O}} / \beta), \quad (5)$$

where β is a coefficient (usually set to 1), $\tilde{m}_{\mathrm{C}}$ is the number of subchannels, and $\exp(\cdot)$ is the exponential function. Similarly, the overall SINR $\Upsilon_{i,j}^{\mathbf{A}}$ of u_i in an ABS is computed by Eq. (5) through replacing $\Upsilon_{i,j,k}^{\mathbf{O}}$ with $\Upsilon_{i,j,k}^{\mathbf{A}}$.

Table 2 lists the MCS supported by each CQI and its minimum required SINR [28], where QPSK and QAM are the acronyms of "quadrature phase-shift keying" and "quadrature amplitude modulation", respectively, and the numbers in brackets indicate code rates. Given $\Upsilon_{i,j}^{\mathbf{O}}$, we can compute the CQI for u_i in a non-ABS (denoted by $Q_i^{\mathbf{O}}$) by finding the maximum CQI whose SINR is no larger than $\Upsilon_{i,j}^{\mathbf{O}}$ from Table 2. The CQI $Q_i^{\mathbf{A}}$ of u_i in an ABS can be found by $\Upsilon_{i,j}^{\mathbf{A}}$ in the same way. Lemma 1 analyzes the time complexity of our CQI evaluation scheme.

Lemma 1. *Given $\tilde{m}_{\mathrm{U}}$ UEs, $\tilde{m}_{\mathrm{E}}$ eNBs, and $\tilde{m}_{\mathrm{C}}$ subchannels, the CQI evaluation scheme has time complexity of $O(\tilde{m}_{\mathrm{U}} \tilde{m}_{\mathrm{E}} \tilde{m}_{\mathrm{C}})$ in the worst case.*

Proof: We first compute P^{ABS} and P_i^{LC} by Eq. (2), which finds the minimum delay budget of all flows. Obviously, it takes $O(\tilde{m}_{\mathrm{F}})$ time, where $\tilde{m}_{\mathrm{F}}$ is the number of flows. Then, Eq. (3) finds the SINR of a UE over a subchannel, which spends $O(\tilde{m}_{\mathrm{E}})$ time as we should consider the power $S(u_i, e_h, c_k)$ from every eNB e_h. Since there are $\tilde{m}_{\mathrm{U}}$ UEs and $\tilde{m}_{\mathrm{C}}$ subchannels, it takes time of $\tilde{m}_{\mathrm{U}} \cdot \tilde{m}_{\mathrm{C}} \cdot O(\tilde{m}_{\mathrm{E}})$ to find $\Upsilon_{i,j,k}^{\mathbf{O}}$ for all UEs in a non-ABS. On the other hand, we need not find SINRs of macrocell UEs in an ABS. Besides, it spends time of $\tilde{m}_{\mathrm{U,P}} \cdot \tilde{m}_{\mathrm{C}} \cdot O(\tilde{m}_{\mathrm{E,P}})$ to find $\Upsilon_{i,j,k}^{\mathbf{A}}$ of all picocell UEs in an ABS (since Eq. (4) considers the power only from picocell eNBs), where $\tilde{m}_{\mathrm{U,P}}$ and $\tilde{m}_{\mathrm{E,P}}$ denote the numbers of picocell UEs and eNBs, respectively.

After getting all $\Upsilon_{i,j,k}^{\mathbf{O}}$ (respectively, $\Upsilon_{i,j,k}^{\mathbf{A}}$) values, we use EESM in Eq. (5) to find $\Upsilon_{i,j}^{\mathbf{O}}$ (respectively, $\Upsilon_{i,j}^{\mathbf{A}}$) for each UE, which takes $O(\tilde{m}_{\mathrm{C}})$ time to do the calculation. As there are $\tilde{m}_{\mathrm{U}}$ UEs, it spends time of $2\tilde{m}_{\mathrm{U}} \cdot O(\tilde{m}_{\mathrm{C}})$ to estimate the overall SINR of each UE. Finally, we pick a CQI for each UE in non-ABSs and ABSs by consulting Table 2. Since Table 2 contains only 15 CQIs, it takes constant time to do the consultation. Thus, finding CQIs for all UEs takes $2O(\tilde{m}_{\mathrm{U}})$ time.

To sum up, the CQI evaluation scheme has time complexity of $O(\tilde{m}_{\mathrm{F}}) + \tilde{m}_{\mathrm{U}} \cdot \tilde{m}_{\mathrm{C}} \cdot O(\tilde{m}_{\mathrm{E}}) + \tilde{m}_{\mathrm{U,P}} \cdot \tilde{m}_{\mathrm{C}} \cdot O(\tilde{m}_{\mathrm{E,P}}) +$

$2\tilde{m}_{\mathrm{U}} \cdot O(\tilde{m}_{\mathrm{C}}) + 2O(\tilde{m}_{\mathrm{U}})$. Since $\tilde{m}_{\mathrm{U,P}} \leq \tilde{m}_{\mathrm{U}}$, $\tilde{m}_{\mathrm{E,P}} \leq \tilde{m}_{\mathrm{E}}$, and $\tilde{m}_{\mathrm{F}} \leq \tilde{m}_{\mathrm{U}}\tilde{m}_{\mathrm{C}}\tilde{m}_{\mathrm{E}}$ (unless each UE has more than $\tilde{m}_{\mathrm{C}}\tilde{m}_{\mathrm{E}}$ flows, but it is infeasible), we can simplify the above equation to $O(\tilde{m}_{\mathrm{U}}\tilde{m}_{\mathrm{E}}\tilde{m}_{\mathrm{C}})$ and verify this lemma. □

4.2.2 Resource Estimation

This stage estimates the amount of resources (in PRBs) required by each UE. LTE-A offers three tables [29] to find the number of data bits that a UE can get based on its CQI and PRBs:

- *CQI-MCS mapping table:* This table provides the translation of a CQI Q_i into an MCS index I_{MCS}. We use a function $F_1(Q_i) = I_{\mathrm{MCS}}$ to represent the translation.
- *MCS-TBS mapping table:* It translates I_{MCS} into a TBS (transport block size) index I_{TBS}, where TBS gives the number of bits that a PRB can carry. This translation is described by a function $F_2(I_{\mathrm{MCS}}) = I_{\mathrm{TBS}}$.
- *TBS-bit mapping table:* Given I_{TBS} and n_i PRBs, the table returns how many bits (i.e., b_i) carried by these PRBs. A function $F_3(I_{\mathrm{TBS}}, n_i) = b_i$ is used to indicate the translation.

Through these tables, we can compute the number of PRBs used to send data with b_i bits:

$$F(b_i, Q_i) = \arg\min_{n_i}\{F_3(F_2(F_1(Q_i)), n_i) \geq b_i\}. \quad (6)$$

Eq. (6) means to use the minimum PRBs that carry at least b_i bits based on CQI Q_i.

Suppose that each flow $f_{i,j}$ of UE u_i has data rate of $r_{i,j}$ (in bps). We compute the number of PRBs to satisfy u_i's GBR demands in a cycle with non-ABSs:

$$n_i^{\mathbf{G,O}} = \sum\nolimits_{\forall f_{i,j}} \{F(r_{i,j} \cdot P_i^{\mathrm{LC}}/1000, Q_i^{\mathbf{O}}) \,|\, f_{i,j} \text{ is GBR}\}, \quad (7)$$

the number of PRBs to satisfy u_i's GBR demands in a cycle with ABSs:

$$n_i^{\mathbf{G,A}} = \sum\nolimits_{\forall f_{i,j}} \{F(r_{i,j} \cdot P_i^{\mathrm{LC}}/1000, Q_i^{\mathbf{A}}) \,|\, f_{i,j} \text{ is GBR}\}, \quad (8)$$

the number of PRBs to meet u_i's non-GBR demands in a cycle with non-ABSs:

$$n_i^{\mathbf{NG,O}} = \sum\nolimits_{\forall f_{i,j}} \{F(r_{i,j} \cdot P_i^{\mathrm{LC}}/1000, Q_i^{\mathbf{O}}) \,|\, f_{i,j} \text{ is non-GBR}\}, \quad (9)$$

and the number of PRBs to meet u_i's non-GBR demands in a cycle with ABSs:

$$n_i^{\mathbf{NG,A}} = \sum\nolimits_{\forall f_{i,j}} \{F(r_{i,j} \cdot P_i^{\mathrm{LC}}/1000, Q_i^{\mathbf{A}}) \,|\, f_{i,j} \text{ is non-GBR}\}. \quad (10)$$

In Eqs. (7)–(10), the unit of P_i^{LC} is ms, so $(r_{i,j} \cdot P_i^{\mathrm{LC}}/1000)$ gives the number of data bits produced by $f_{i,j}$ in a cycle. Lemma 2 analyzes the time complexity of the resource estimation scheme.

Lemma 2. *Given $\tilde{m}_{\mathrm{F}}$ flows, the resource estimation scheme spends at most $2O(\tilde{m}_{\mathrm{F}})$ time.*

Proof: Each mapping table can be implemented by an array. In this way, we can easily retrieve the value of any desired entry from a table by giving its index(es) (i.e., input(s)). Thus, the calculation of $F_3(F_2(F_1(Q_i)), n_i)$ takes only constant time. Besides, it is trivial to do the comparison in Eq. (6). So, the calculation of function $F(b_i, Q_i)$ is $O(1)$. Observing from Eqs. (7)–(10), we execute function $F(b_i, Q_i)$ twice for each flow (one for the non-ABS case and the other for the ABS case). Since there are $\tilde{m}_{\mathrm{F}}$ flows, the resource estimation scheme spends time of $2O(\tilde{m}_{\mathrm{F}})$ in the worst case. □

4.2.3 ABS Setting

Let γ be the number of ABSs in a cycle. We set $\gamma = 0$ and estimate the load of each eNB in terms of GBR flows. Then, γ is gradually increased until the load meets some conditions, and we decide the amount of resources given to each GBR flow. If some PRBs remain, they are allocated to non-GBR flows. Specifically, the ABS setting scheme contains five steps below.

Step 1: We compute the number of available PRBs given by each eNB e_j in a cycle:

$$N_j = \begin{cases} \hat{N}(B_j) \cdot P^{\mathrm{ABS}} & \text{if } e_j \in \mathcal{E}_p \\ \hat{N}(B_j) \cdot \max\{P^{\mathrm{ABS}} - \gamma, 0\} & \text{otherwise,} \end{cases} \quad (11)$$

where $\hat{N}(B_j)$ is the number of PRBs that e_j offers in a subframe, which depends on its channel bandwidth B_j (referring to Section 2.1). If e_j is the macrocell eNB, it can allocate PRBs only in $(P^{\mathrm{ABS}} - \gamma)$ non-ABSs. Since it is impossible to use more than P^{ABS} ABSs in a cycle (i.e., $\gamma > P^{\mathrm{ABS}}$), we take the term $\max\{P^{\mathrm{ABS}} - \gamma, 0\}$ in Eq. (11). Then, the load of a macrocell eNB e_m with respect to GBR flows is calculated by

$$L_m = \begin{cases} 0 & \text{if } \mathcal{U}_m = \emptyset \\ \sum_{u_i \in \mathcal{U}_m} n_i^{\mathbf{G,O}}/N_m & \text{if } N_m > 0 \\ L_{\max} & \text{otherwise.} \end{cases} \quad (12)$$

In Eq. (12), if e_m serves no UE (i.e., $\mathcal{U}_m = \emptyset$), its load is obviously zero. On the other hand, if $N_m = 0$ (this case occurs when $\gamma \geq P^{\mathrm{ABS}}$ in Eq. (11)), L_m is set to the maximum limit $L_{\max}$ of load, because e_m cannot send user data in the current cycle.

For a picocell eNB e_j, its load with respect to GBR flows is computed by

$$L_j = (\sum\nolimits_{u_i \in \mathcal{U}_j} n_i^{\mathbf{G,O}} - N_j^{\mathrm{ABS}})/N_j. \quad (13)$$

Since e_j can benefit from ABSs to speed up data transmissions (as channel quality is improved), we deduct N_j^{ABS} from its required number of PRBs in Eq. (13), where

$$N_j^{\mathrm{ABS}} = \sum\nolimits_{u_i \in \mathcal{U}_j} \left\lfloor (n_i^{\mathbf{G,O}}/n_i^{\mathbf{G,A}}) \times \alpha_i^{\mathbf{A}} \right\rfloor - \alpha_i^{\mathbf{A}}. \quad (14)$$

Here, u_i is given with $\alpha_i^{\mathbf{A}}$ ABS-based PRBs[3]. For example, suppose that u_i spends 120 and 20 PRBs to complete sending its GBR data in non-ABSs and ABSs, respectively (i.e., $n_i^{\mathbf{G,O}} = 120$ and $n_i^{\mathbf{G,A}} = 20$). If u_i is given with 10 ABS-based PRBs, it requires extra 60 non-ABS-based PRBs to finish sending its GBR data (i.e., u_i totally spends $10+60 = 70$ PRBs). Thus, comparing with the case where u_i uses only non-ABS-based PRBs, it can save $120 - 70 = 50$ PRBs. From Eq. (14), we have $\lfloor \frac{120}{20} \times 10 \rfloor - 10 = 50$, which derives the same result. Moreover, $\alpha_i^{\mathbf{A}}$ is estimated by

$$\alpha_i^{\mathbf{A}} = \left\lfloor \gamma \hat{N}(B_j)/|\mathcal{U}_j^{\mathbf{G}}| \right\rfloor, \quad (15)$$

which means that we give the average number of ABS-based PRBs to each UE with GBR flows (denoted by $\mathcal{U}_j^{\mathbf{G}}$). In case that $\gamma \hat{N}(B_j)$ is not divisible by $|\mathcal{U}_j^{\mathbf{G}}|$ in Eq. (15), we give one ABS-based PRB to each UE in a round-robin manner.

3. An *ABS-based PRB* is a PRB allocated by one picocell eNB in an ABS.

Step 2: We then determine whether to add ABSs by

$$\sum\nolimits_{e_j \in \mathcal{E}_p} L_j/|\mathcal{E}_p| > 1. \quad (16)$$

Eq. (16) implies that most picocell eNBs have not enough resources to meet their GBR demands, so it is better to use more ABSs to improve picocell throughput. Therefore, we iteratively increase γ by one and recompute each eNB's load by Eqs. (12) and (13), until 1) Eq. (16) is violated, 2) γ reaches P^{ABS} (i.e., no more ABSs can be added), or 3) the condition holds:

$$L_m > 0 \text{ and } \sum\nolimits_{e_j \in \mathcal{E}_p} L_j/|\mathcal{E}_p| < L_m. \quad (17)$$

Eq. (17) means that the macrocell has a heavier load (i.e., L_m) than picocells due to the decrease of N_m in Eq. (12). Thus, we stop adding ABSs to balance loads of the macrocell and its picocells.

Step 3: When $L_m \leq 1$, the macrocell eNB e_m has enough resources to satisfy all GBR demands. Thus, e_m gives each UE its desired number $n_i^{\mathbf{G,O}}$ of PRBs and updates N_m by $N_m - \sum_{u_i \in \mathcal{U}_m} n_i^{\mathbf{G,O}}$ (i.e., N_m is the number of residual PRBs).

When $L_m > 1$, we use a *top-50%-first rule* to assign PRBs to each UE as follows: Let $Q_{\mathrm{avg}}^{\mathbf{O}}$ be the average CQI. UEs are divided into *good-channel* and *bad-channel* groups (denoted by $\hat{G}_{\mathrm{g}}$ and $\hat{G}_{\mathrm{b}}$, respectively). If a UE u_i satisfies the condition of $Q_i^{\mathbf{O}} \geq \max\{Q_{\mathrm{avg}}^{\mathbf{O}}, Q_{\mathrm{th}}\}$, where Q_{th} is a lower-bound threshold for CQI, it is added to $\hat{G}_{\mathrm{g}}$. Otherwise, u_i is added to $\hat{G}_{\mathrm{b}}$. Then, we first allocate PRBs to the UEs in $\hat{G}_{\mathrm{g}}$. Specifically, we repeat the following operations: 1) pick a UE u_i with the largest CQI from $\hat{G}_{\mathrm{g}}$, 2) allocate $n_i^{\mathbf{G,O}}$ PRBs to u_i, and 3) set $N_m = N_m - n_i^{\mathbf{G,O}}$, until $N_m = 0$ or $\hat{G}_{\mathrm{g}} = \emptyset$. If $N_m > 0$ (i.e., some PRBs remain), we proportionally distribute them among the UEs in $\hat{G}_{\mathrm{b}}$. Specifically, for each UE $u_i \in \hat{G}_{\mathrm{b}}$, we give it $\lfloor (n_i^{\mathbf{G,O}} / \sum_{u_k \in \hat{G}_{\mathrm{b}}} n_k^{\mathbf{G,O}}) \times N_m \rfloor$ PRBs. Remark 2 discusses the idea behind the top-50%-first rule.

Step 4: A picocell eNB e_j has sufficient resources to meet all GBR demands if $L_j \leq 1$. In this case, e_j gives the required number of PRBs to each UE as follows:

- Sort all UEs by their $Q_i^{\mathbf{A}}$ values in a decreasing order.
- Based on the order, we iteratively allocate a number $n_i^{\mathbf{G,A}}$ of ABS-based PRBs to a UE u_i to meet its GBR demand, until we use up ABS-based PRBs. However, if there are not enough ABS-based PRBs to satisfy u_i's demand, we adopt Eq. (14) to compute the number of extra non-ABS-based PRBs allocated to u_i.
- Then, we give a number $n_i^{\mathbf{G,O}}$ of non-ABS-based PRBs to each UE in $\mathcal{U}_j$ whose GBR demand has not been satisfied yet. Also, N_j is updated by the number of residual PRBs owned by e_j.

Otherwise, e_j cannot support GBR demands of all UEs. Thus, e_j adopts the top-50%-first rule to assign PRBs to its UEs, and then sets $N_j = 0$.

Step 5: If $N_j \neq 0$ for some eNBs (i.e., they still have available PRBs), we check whether N_j is large enough to meet $n_i^{\mathbf{NG,O}}$ or $n_i^{\mathbf{NG,A}}$ demands of non-GBR flows in their cells. If so, these UEs are given with PRBs as they require (i.e., similar to the above method to deal with GBR flows). Otherwise, we use the top-50%-first rule to assign PRBs to UEs for their non-GBR flows.

By the above steps, the ABS ratio will be $\delta = \gamma / P^{\mathrm{ABS}}$. Remark 1 discusses why the ABS setting scheme can find a suitable γ value to balance loads of a macrocell and its picocells with respect to GBR flows. Then, Lemma 3 analyzes the time complexity of this scheme.

Remark 1 (Load-balancing property)**.** *Our ABS setting scheme can balance GBR loads of a macrocell and its picocells (i.e., minimizing the difference between their GBR loads). Specifically, there are three possible cases for a macrocell eNB e_m and its picocell eNBs $e_j \in \mathcal{E}_p$ in the beginning: 1) $L_m = 0$, 2) $\sum_{e_j \in \mathcal{E}_p} L_j/|\mathcal{E}_p| < L_m$, and 3) $\sum_{e_j \in \mathcal{E}_p} L_j/|\mathcal{E}_p| \geq L_m$.*

For case 1, since e_m has no GBR load, it does not matter to balance loads of e_m and picocell eNBs. However, the ABS setting scheme still increases γ until either $\sum_{e_j \in \mathcal{E}_p} L_j/|\mathcal{E}_p| \leq 1$ (i.e., each picocell eNB can satisfy all GBR demands) or $\gamma = P^{\mathrm{ABS}}$ (i.e., no ABSs can be further added). In this way, we guarantee that the overall GBR throughput is maximized.

Case 2 implies that e_m has a heavier GBR load than its picocell eNBs, even if there is no ABS in the current cycle. In this case, increasing γ will raise the gap between the load of e_m and those of its picocell eNBs, because L_m increases (referring to Eqs. (11) and (12)) while $\sum_{e_j \in \mathcal{E}_p} L_j/|\mathcal{E}_p|$ decreases (referring to Eq. (13)) when γ grows. In fact, the condition of Eq. (17) holds in the beginning, which makes the ABS setting scheme keep $\gamma = 0$ and minimize the gap accordingly.

For case 3, the ABS setting scheme starts from $\gamma = 0$ and increases γ by one in each iteration of step 2. When the GBR loads of e_m and its picocells become balanced (i.e., the gap is minimized), the condition of Eq. (17) will also become true. In this case, the ABS setting scheme stops increasing γ, which gets a suitable γ value to balance cell loads. □

Lemma 3. *Let $\tilde{m}_{\mathrm{U}}$, $\tilde{m}_{\mathrm{E}}$, and $\tilde{m}_{\mathrm{F}}$ be the numbers of UEs, eNBs, and flows, respectively. Then, the worst-case time complexity of the ABS setting scheme is $O(P^{\mathrm{ABS}} \cdot \tilde{m}_{\mathrm{E}} \tilde{m}_{\mathrm{U}}) + O(\tilde{m}_{\mathrm{F}} \lg \tilde{m}_{\mathrm{F}})$.*

Proof: The ABS setting scheme can be divided into two parts. Part 1 repeats both steps 1 and 2 to calculate γ. Part 2 contains steps 3, 4, and 5 for each eNB to allocate PRBs to its UEs. Below, we analyze the time complexity of each part.

In an iteration of part 1, we use Eq. (12) to find the GBR load of each macrocell, which spends time of $\tilde{m}_{\mathrm{E,M}} \cdot O(\tilde{m}_{\mathrm{U,M}})$, where $\tilde{m}_{\mathrm{E,M}}$ and $\tilde{m}_{\mathrm{U,M}}$ denote the numbers of macrocell eNBs and UEs, respectively. Similarly, we use Eq. (13) to estimate the GBR load of each picocell, which takes time of $\tilde{m}_{\mathrm{E,P}} \cdot O(2\tilde{m}_{\mathrm{U,P}})$, where $\tilde{m}_{\mathrm{E,P}}$ and $\tilde{m}_{\mathrm{U,P}}$ are the numbers of picocell eNBs and UEs, respectively. Then, we check whether to add ABSs by Eq. (16), which requires $O(\tilde{m}_{\mathrm{E,P}})$ time. The three conditions in step 2 actually take constant time (including the calculation of Eq. (17), as we already know the value of $\sum_{e_j \in \mathcal{E}_p} L_j/|\mathcal{E}_p|$ from Eq. (16)). Thus, one iteration takes time of $\tilde{m}_{\mathrm{E,M}} \cdot O(\tilde{m}_{\mathrm{U,M}}) + \tilde{m}_{\mathrm{E,P}} \cdot O(2\tilde{m}_{\mathrm{U,P}}) + O(\tilde{m}_{\mathrm{E,P}}) = O(\tilde{m}_{\mathrm{E,M}} \cdot \tilde{m}_{\mathrm{U,M}} + \tilde{m}_{\mathrm{E,P}} \cdot \tilde{m}_{\mathrm{U,P}})$. The worst case occurs when we increase γ from zero to P^{ABS}. In other words, there are at most P^{ABS} iterations. So, the time complexity of part 1 is $P^{\mathrm{ABS}} \cdot O(\tilde{m}_{\mathrm{E,M}} \cdot \tilde{m}_{\mathrm{U,M}} + \tilde{m}_{\mathrm{E,P}} \cdot \tilde{m}_{\mathrm{U,P}})$. Since $\tilde{m}_{\mathrm{E,M}} \leq \tilde{m}_{\mathrm{E}}$, $\tilde{m}_{\mathrm{E,P}} \leq \tilde{m}_{\mathrm{E}}$, $\tilde{m}_{\mathrm{U,M}} \leq \tilde{m}_{\mathrm{U}}$, and $\tilde{m}_{\mathrm{U,P}} \leq \tilde{m}_{\mathrm{U}}$, we can simplify the above equation to $O(P^{\mathrm{ABS}} \cdot \tilde{m}_{\mathrm{E}} \tilde{m}_{\mathrm{U}})$.

For part 2, we observe that the top-50%-first rule dominates the computational time of steps 3, 4, and 5. Thus, the worst case is that all flows belong to GBR and an eNB has to use this rule to allocate PRBs to every flow. In the top-50%-first rule, we sort flows by their CQIs, which spends $O(\tilde{m}_{\mathrm{F}} \lg \tilde{m}_{\mathrm{F}})$ time. Then, the eNB iteratively picks one flow and gives PRBs to it.

This operation takes $O(\tilde{m}_{\mathrm{F}})$ time. Thus, the time complexity of part 2 is $O(\tilde{m}_{\mathrm{F}} \lg \tilde{m}_{\mathrm{F}}) + O(\tilde{m}_{\mathrm{F}}) = O(\tilde{m}_{\mathrm{F}} \lg \tilde{m}_{\mathrm{F}})$.

To sum up, the time complexity of the ABS setting scheme is $O(P^{\mathrm{ABS}} \cdot \tilde{m}_{\mathrm{E}} \tilde{m}_{\mathrm{U}}) + O(\tilde{m}_{\mathrm{F}} \lg \tilde{m}_{\mathrm{F}})$, which verifies this lemma. □

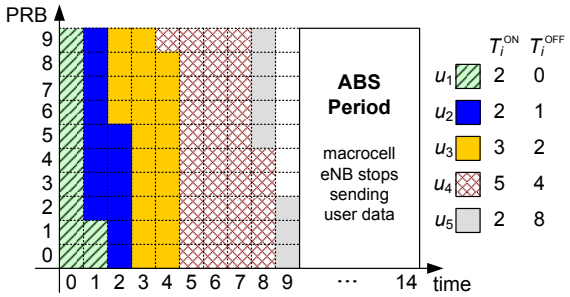
Remark 2 (The idea behind the top-50%-first rule)**.** *Both* max-CQI *and* proportional fair (PF) *are two classic scheduling policies in LTE-A [30]. Max-CQI always gives each PRB to the UE with the best channel quality (i.e., the largest CQI). Due to its greedy nature, max-CQI can achieve the highest throughput in theory. However, other UEs with smaller CQIs may not get any resource (i.e., starvation). On the contrary, PF seeks to allocate PRBs in a fair manner, so UEs may get PRBs proportionally to their demands. However, network throughput will significantly degrade if some UEs have pretty low CQIs. Based on these observations, our top-50%-first rule combines the advantages of both max-CQI and PF policies. In particular, it divides UEs into good-channel $\hat{G}_g$ and bad-channel $\hat{G}_b$ groups. Group $\hat{G}_g$ contains the first 50% of UEs in terms of channel quality. Thus, our rule adopts the max-CQI policy to allocate PRBs to these UEs, which can greatly improve throughput by using fewer PRBs. On the other hand, if there are residual PRBs, they are given to other UEs (i.e., $\hat{G}_b$) following the PF policy, so as to avoid starving some UEs. Note that when most UEs have bad channel quality, it is not suitable to adopt the max-CQI policy [31]. The reason is that each UE spends more PRBs to carry its data. Since the total number of PRBs is fixed, max-CQI starves more UEs. In this case, PF is a better choice. Therefore, we apply the lower-bound threshold Q_{th} to the top-50%-first rule. When $Q_i^{\mathbf{O}} \geq Q_{\mathrm{avg}}^{\mathbf{O}}$ but $Q_i^{\mathbf{O}} < Q_{\mathrm{th}}$, UE u_i is added to the bad-channel group $\hat{G}_b$, which is allocated with PRBs by the PF policy. The work [32] points out that a PRB carries relatively more data bits by using QAM than using QPSK. Thus, we suggest setting Q_{th} to the lowest CQI capable of using QAM (i.e., $Q_{\mathrm{th}} = 7$ from Table 2). Based on the above design, the top-50%-first rule can keep high throughput even if there are few PRBs available (thanks to the max-CQI property). Besides, when there are more PRBs offered by an eNB, this rule can improve UE fairness by taking the PF policy.* □

4.2.4 DRX Setting

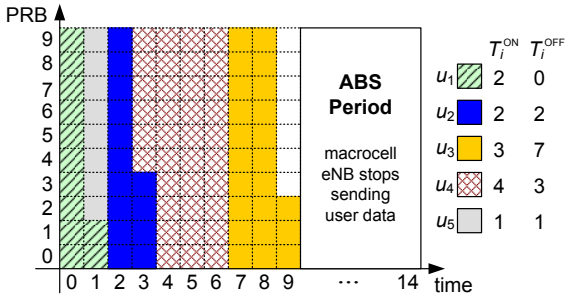
After determining the number of PRBs given to each UE u_i, we can find its *PRB deployment* and also T_i^{ON} and T_i^{OFF} values. To better utilize the spectrum resource and let UEs save more energy, we should compact PRBs allocated to them. There are two cases to be discussed. For the macrocell case, we start deploying PRBs from the beginning of an ABS cycle, and place a number $(n_i^{\mathbf{G},\mathbf{O}} + n_i^{\mathbf{NG},\mathbf{O}})$ of PRBs for each UE in sequence. Fig. 3(a) gives an example, where $P^{\mathrm{ABS}} = 15\,\mathrm{ms}$ and $\hat{N}(B_j) = 10$. Suppose that UEs u_1, u_2, u_3, u_4, and u_5 are given with 12, 14, 23, 36, and 8 PRBs by the ABS setting scheme, respectively. From the deployment of PRBs in Fig. 3(a), we can easily decide both T_i^{ON} and T_i^{OFF} for each UE.

For the picocell case, we first deploy PRBs for the UEs which use only non-ABS-based PRBs (i.e., $n_i^{\mathbf{G},\mathbf{A}} + n_i^{\mathbf{NG},\mathbf{A}} = 0$ but $n_i^{\mathbf{G},\mathbf{O}} + n_i^{\mathbf{NG},\mathbf{O}} > 0$) by the method in the macrocell case. Then, we deploy PRBs for the UEs which use both non-ABS-based and ABS-based PRBs (i.e., $n_i^{\mathbf{G},\mathbf{A}} + n_i^{\mathbf{NG},\mathbf{A}} > 0$ and $n_i^{\mathbf{G},\mathbf{O}} + n_i^{\mathbf{NG},\mathbf{O}} > 0$). Finally, we deploy PRBs for the UEs which use merely ABS-based PRBs (i.e., $n_i^{\mathbf{G},\mathbf{A}} + n_i^{\mathbf{NG},\mathbf{A}} > 0$ but $n_i^{\mathbf{G},\mathbf{O}} + n_i^{\mathbf{NG},\mathbf{O}} = 0$). In this way, we can guarantee that all UEs are synchronized by the same ABS cycle.

However, the above deployment of PRBs is not optimal, since some UEs have to keep awake for a longer T_i^{ON} time, thereby spending more energy. Fig. 3 gives two examples. When we deploy PRBs based on the sequence of UEs, as shown in Fig. 3(a), all UEs wake up for 14 subframes. In fact, one can use the deployment of PRBs in Fig. 3(b) to minimize the awake time to 12 subframes. We formulate this problem as a *PRB deployment with the minimum awake subframes (PDMAS) problem* in Definition 1, and Theorem 1 shows that it is NP-hard.



(a) deployment by UE sequence



(b) optimal deployment

Fig. 3: Two examples of PRB deployment in the macrocell case.

Definition 1. *Given the number of PRBs $\hat{N}(B_j)$ provided by eNB e_j in a subframe and the number of PRBs n_i assigned to each UE u_i, the PDMAS problem asks how to arrange the PRBs allocated to UEs in a cycle, such that the sum of T_i^{ON} values of all UEs in $\mathcal{U}_j$ is minimized.*

Theorem 1. *The PDMAS problem is NP-hard.*

Proof: Observing from Fig. 3, when n_i is divisible by $\hat{N}(B_j)$ for a UE u_i, the best way to minimize its T_i^{ON} value is to give it the *whole* PRBs in $n_i/\hat{N}(B_j)$ subframes. Thus, u_i will not share the PRBs in a subframe with others. In general, when a subset $\tilde{\mathcal{U}}_j \subset \mathcal{U}_j$ of UEs meet the condition of $\sum_{u_i \in \tilde{\mathcal{U}}_j} n_i \mod \hat{N}(B_j) = 0$, we can allocate the whole PRBs in $\sum_{u_i \in \tilde{\mathcal{U}}_j} n_i/\hat{N}(B_j)$ subframes to minimize their T_i^{ON} values. Fig. 3(b) gives an example, where we put the PRBs of u_1 and u_5 in subframes 0 and 1, and also the PRBs of u_2 and u_4 in subframes 2–6. Thus, if we find all of such subsets $\tilde{\mathcal{U}}_j$, the sum of T_i^{ON} values of all UEs must be the minimum.

Since any two UEs in different subsets will not share PRBs in the same subframe, the PDMAS problem will be equivalent to the problem of finding the largest subset $\tilde{\mathcal{U}}_j$ of UEs such that $\sum_{u_i \in \tilde{\mathcal{U}}_j} n_i \mod \hat{N}(B_j) = 0$. Thus, we formulate its decision problem as follows: Given n_i of each UE in $\mathcal{U}_j$, is there any subset of UEs such that $\sum_{u_i \in \tilde{\mathcal{U}}_j} n_i = k\hat{N}(B_j)$, where $k \in \mathbb{N}$ is a constant. In fact, there exists a similar NP-complete problem, called the *subset-sum problem*: Given a set $\mathcal{X}$ of integers and also a target integer t, can we find a non-empty subset from $\mathcal{X}$ whose sum is equal to t? For example, given $\mathcal{X} = \{3, 1, 7, 8, 10\}$ and $t = 16$, the solution is $\{1, 7, 8\}$. Thus,

we construct a PDMAS problem instance to do the reduction from the subset-sum problem. Suppose that $n_i < \hat{N}(B_j)$ for any UE in $\mathcal{U}_j = \{u_1, u_2, \cdots, u_m\}$. Then, we construct a set $\mathcal{X} = \{x_1, x_2, \cdots, x_m\}$ such that $x_i = n_i$, for $i = 1..m$. In this way, if we find a solution to the subset-sum problem, we can derive the solution subset $\widetilde{\mathcal{U}}_j$ to the PDMAS problem, and vice versa. The above reduction takes polynomial time, so the PDMAS problem is NP-hard. □

To solve the subset-sum problem, we adopt the linear approximation algorithm in [33], whose idea is to split the set $\mathcal{X}$ into two subsets that consist of large and small integers. It selects large integer(s) and then adds small integers based on a greedy approach, which checks small integers in any order and inserts each new one if the sum does not exceed t. Given the set $\mathcal{X}$ and the target integer t, this algorithm contains four steps to find the solution list $\mathcal{L}$:

1. Let s_1 be the sum of the three minimum integers greater than $\lceil t/4 \rceil$. If $s_1 \leq t$, we set $s = s_1$, move the three integers from $\mathcal{X}$ to $\mathcal{L}$, and go to step 4.
2. Let s_2 be the sum of the minimum integer greater than $\lceil t/4 \rceil$ and the minimum integer greater than $\lceil t/2 \rceil$. If $s_2 \leq t$, we set $s = s_2$, move the two integers to $\mathcal{L}$, and go to step 4.
3. Let s_3 be the sum of the two largest integers which do not exceed $\lceil t/2 \rceil$ and also s_4 be the largest integer. We set $s = \max\{s_3, s_4\}$, and move the corresponding integer(s) to $\mathcal{L}$ based on the selection of $\max$ operation.
4. Iteratively pick integers from $\mathcal{X}$ by the greedy approach and move them to $\mathcal{L}$, until their sum reaches $(t - s)$.

Let us use the term $\mathcal{L} = \mathrm{LA}(\mathcal{X}, t)$ to denote the above algorithm. Then, we propose a method to solve the PDMAS problem as follows:

1. For each UE $u_i \in \mathcal{U}_j$, we check if $n_i \bmod \hat{N}(B_j) = 0$. If so, we give n_i PRBs to u_i and remove it from $\mathcal{U}_j$. When $\mathcal{U}_j = \emptyset$, the method finishes as we have deployed PRBs for all UEs.
2. Build a set $\mathcal{X}$ of integers from $\mathcal{U}_j$, where each integer $x_i \in \mathcal{X}$ is computed by $(n_i \mod \hat{N}(B_j))$ corresponding to each UE $u_i \in \mathcal{U}_j$. Then, we run $\mathrm{LA}(\mathcal{X}, \hat{N}(B_j))$ and get the list $\mathcal{L}$. If $\sum_{x_i \in \mathcal{L}} x_i = \hat{N}(B_j)$, we deploy PRBs for the corresponding UEs by referring to $\mathcal{L}$, and remove these UEs from $\mathcal{U}_j$. Otherwise, we go to step 4.
3. Repeat step 2, until $|\mathcal{U}_j| < 3$ (since it is a trivial case when there exist only two UEs).
4. Deploy PRBs for residual UEs in $\mathcal{U}_j$ in sequence.

Fig. 3 gives an example, where $\hat{N}(B_j) = 10$. In step 1, as no single UE can occupy the whole PRBs in an integral number of subframes, we skip this step. In step 2, we construct a set $\mathcal{X} = \{2, 4, 3, 6, 8\}$ based on the n_i value of each UE. Given $\lceil \hat{N}(B_j)/4 \rceil = 3$ and $\lceil \hat{N}(B_j)/2 \rceil = 5$, we run $\mathrm{LA}(\mathcal{X}, \hat{N}(B_j))$ as follows: 1) $s_1 = 4 + 6 + 8 = 18 > \hat{N}(B_j)$, and the condition does not hold; 2) $s_2 = 4 + 6 \leq \hat{N}(B_j)$, so $s = s_2$ and $\mathcal{L} = \{4, 6\}$; 3) since $\hat{N}(B_j) - s = 0$, we return $\mathcal{L}$. Thus, we first deploy PRBs for u_2 and u_4, followed by the PRBs of u_1 and u_5 (by step 2 again). Then, since $\mathcal{U}_j$ has only u_3, we deploy n_3 PRBs for it. Theorem 2 shows that our PDMAS method has an approximation ratio of 3/4 to the optimal solution, and Lemma 4 analyzes the time complexity of the DRX setting scheme.

Theorem 2. *The PDMAS method is a 3/4-approximation algorithm.*

Proof: The work [33] proves that the $\mathrm{LA}(\mathcal{X}, t)$ algorithm is a 3/4-approximation algorithm to solve the subset-sum problem. In Theorem 1, we also verify that the subset-sum problem can be reduced to the PDMAS problem. As the PDMAS method adopts the $\mathrm{LA}(\mathcal{X}, t)$ algorithm to find the solution, it is a 3/4-approximation algorithm. □

Lemma 4. *Given $\tilde{m}_{\mathrm{U}}$ UEs, the DRX setting scheme spends at most $O(\tilde{m}_{\mathrm{U}}^2)$ time.*

Proof: The DRX setting scheme uses the PDMAS method to arrange PRBs for UEs. In its step 1, we first check if the condition of $n_i \bmod \hat{N}(B_j) = 0$ holds for each UE. This step thus spends $O(\tilde{m}_{\mathrm{U}})$ time. Then, step 2 adopts the $\mathrm{LA}(\mathcal{X}, t)$ algorithm to get the list $\mathcal{L}$, which takes $O(|\mathcal{X}|)$ time [33]. The worst case occurs when $\mathrm{LA}(\mathcal{X}, t)$ returns $\mathcal{L}$ with only two items in each iteration of step 2. Thus, it spends time of $O(\tilde{m}_{\mathrm{U}})$, $O(\tilde{m}_{\mathrm{U}} - 2)$, $O(\tilde{m}_{\mathrm{U}} - 4)$, $\cdots$, and $O(2)$ in the 1st, 2nd, 3rd, $\cdots$, and $(\frac{\tilde{m}_{\mathrm{U}}}{2})$th iterations, respectively. Obviously, this is an arithmetic sequence. So, both steps 2 and 3 take time of

$$\sum_{i=0}^{\frac{\tilde{m}_{\mathrm{U}}}{2}-1} O(\tilde{m}_{\mathrm{U}} - 2i) = O\left(\frac{\tilde{m}_{\mathrm{U}}(\tilde{m}_{\mathrm{U}} + 2)}{4}\right) = O(\tilde{m}_{\mathrm{U}}^2).$$

After deploying PRBs for each UE, we can get its T_i^{ON} and T_i^{OFF} values, which requires to search all UEs once. This operation takes $O(\tilde{m}_{\mathrm{U}})$ time. Therefore, the time complexity of the DRX setting scheme is $O(\tilde{m}_{\mathrm{U}}) + O(\tilde{m}_{\mathrm{U}}^2) + O(\tilde{m}_{\mathrm{U}}) = O(\tilde{m}_{\mathrm{U}}^2)$, which verifies the lemma. □

4.3 Discussion

We give the rationale of our JIPM mechanism. The CQI evaluation scheme in stage 1 not only decides the lengths of ABS and DRX cycles, but also computes SINR of each UE with or without ABS. To meet QoS demands of different flows, we pick the strictest delay budget to set the long DRX cycle. Besides, to allow macrocell UEs going to sleep in ABSs, we also synchronize both DRX and ABS cycles. Based on SINR and cycle length, the resource estimation scheme in stage 2 calculates the number of PRBs used to send out GBR and non-GBR data of each UE in ABSs and non-ABSs.

The ABS setting scheme in stage 3 is the core of JIPM, which adjusts the ABS ratio and finds the actual number of PRBs given to UEs. Although the ABS method facilitates data transmissions in picocells by improving their signal quality, it degrades macrocell throughput. To improve network throughput while supporting fairness, the ABS setting scheme adopts three policies:

- **P1.** GBR flows always have the highest priority to meet their delay requirements.
- **P2.** Traffic loads of a macrocell and its picocells should be balanced.
- **P3.** In each cell, we increase its throughput but should avoid starving many UEs.

For policy P1, we first give PRBs to GBR flows. Only when an eNB has residual PRBs will they be allocated to non-GBR flows (by step 5). Thus, if PRBs are not enough, we can meet QoS demands of GBR flows as much as possible. For policy P2, we start from $\gamma = 0$ and iteratively increase γ by one to reduce loads of picocells, until the condition of Eq. (17) is satisfied

(i.e., load balance). For policy P3, we use the top-50%-first rule to allocate PRBs by giving the desired number of PRBs to the UEs with good channel quality while distributing residual PRBs among other UEs proportionally to their demands. As discussed in Remark 2, the top-50%-first rule is a hybrid of both max-CQI and PF, which can increase throughput and improve fairness.

Finally, the DRX setting scheme in stage 4 decides both T_i^{ON} and T_i^{OFF} of each UE. To synchronize the sleeping period of macrocell UEs with ABSs, we align the deployment of PRBs with the ABS cycle, as Fig. 3 shows. Besides, we deploy non-ABS-based PRBs for picocell UEs, followed by ABS-based PRBs. In this way, we can also synchronize the ABS cycles of macrocell and picocell UEs. Even though different strategies of PRB deployment do not change the total number of PRBs used (referring to the examples in Fig. 3), they will affect the on-duration timer T_i^{ON} of each UE, thereby deciding its energy consumption. Thus, we propose the NP-hard PDMAS problem to minimize T_i^{ON} values of UEs, and develop a 3/4-approximation method. With these designs, the JIPM mechanism can integrate the ABS and DRX methods to jointly manage interference and power in LTE-A HetNets. Theorem 3 analyzes the time complexity of JIPM.

Theorem 3. *Given $\tilde{m}_{\mathrm{U}}$ UEs, $\tilde{m}_{\mathrm{E}}$ eNBs, $\tilde{m}_{\mathrm{F}}$ flows, and $\tilde{m}_{\mathrm{C}}$ subchannels, the JIPM mechanism has time complexity of $O(\tilde{m}_{\mathrm{U}}\tilde{m}_{\mathrm{E}} \cdot (\tilde{m}_{\mathrm{C}} + P^{\mathrm{ABS}}) + \tilde{m}_{\mathrm{F}} \lg \tilde{m}_{\mathrm{F}} + \tilde{m}_{\mathrm{U}}^2)$ in the worst case.*

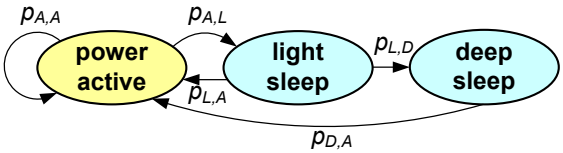
Proof: JIPM comprises the four schemes in Sections 4.2.1–4.2.4. According to Lemmas 1–4, the worst-case time complexity of JIPM is $O(\tilde{m}_{\mathrm{U}}\tilde{m}_{\mathrm{E}}\tilde{m}_{\mathrm{C}}) + 2O(\tilde{m}_{\mathrm{F}}) + O(P^{\mathrm{ABS}} \cdot \tilde{m}_{\mathrm{E}}\tilde{m}_{\mathrm{U}}) + O(\tilde{m}_{\mathrm{F}} \lg \tilde{m}_{\mathrm{F}}) + O(\tilde{m}_{\mathrm{U}}^2) = O(\tilde{m}_{\mathrm{U}}\tilde{m}_{\mathrm{E}} \cdot (\tilde{m}_{\mathrm{C}} + P^{\mathrm{ABS}}) + \tilde{m}_{\mathrm{F}} \lg \tilde{m}_{\mathrm{F}} + \tilde{m}_{\mathrm{U}}^2)$, which verifies the theorem. □

5 Performance Evaluation

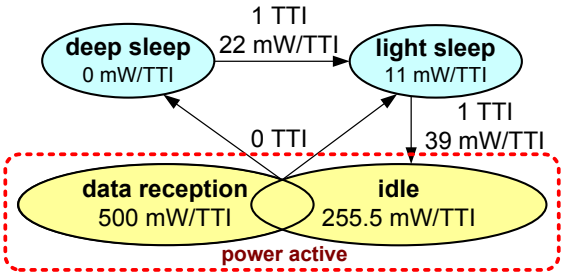
We evaluate the performance of our JIPM mechanism by LTE-Sim, which is an open-source simulation framework to imitate transmission behavior in LTE-A [34]. Below, we introduce standard models for DRX considered by 3GPP, followed by the settings of simulation. Then, we discuss the experimental results, including energy efficiency, network throughput, energy consumption of UEs, GBR packet dropping, and effect of UE distribution.

5.1 Standard Models for DRX

There are two standard models used to describe UE behavior in DRX, namely *3-state* and *Nokia*. The 3-state model is a semi-Markov process that contains the states of *power active*, *light sleep*, and *deep sleep*, as shown in Fig. 4(a). The power-active state comprises a sequence of adjacent active time intervals, which corresponds to the duration of one single packet call. A UE follows DRX short cycles when it stays in the light-sleep state. When a UE enters the deep-sleep state, it actually follows DRX long cycles. If a UE is in the power-active state, it will keep active with a probability $p_{A,A}$ and change to the light-sleep state with a probability $p_{A,L}$. On the other hand, when the UE is in the light-sleep state, it will become awake with a probability $p_{L,A}$ or go to deeply sleep with a probability $p_{L,D}$. Besides, when the UE is in the deep-sleep state, there is a probability $p_{D,A}$ that it will switch to the power-active state. The work [35] gives the analysis of the above state-transition probabilities.



(a) 3-state model



(b) Nokia model

Fig. 4: Two standard models for DRX considered by 3GPP.

TABLE 3: Simulation parameters of eNBs.

parameter	value
channel bandwidth	10 MHz (eNB offers 50 PRBs per TTI)
frame structure	frequency division duplexing (FDD)
modulation	QPSK, 16QAM, 64QAM
transmitted power	macrocell: 46 dBm, picocell: 30 dBm
cell range	macrocell: 1 km, picocell: 100 m
path loss [38]	macrocell: $128.1 + 37.6 \log \zeta$ picocell: $38 + 30 \log \zeta$
shadowing fading	log-normal distribution with 0 dB mean and 8 dB standard deviation
penetration loss	10 dB
fast fading	Jakes model (for Rayleigh fading)
CRE bias	9 dB

The Nokia model [36] further divides the power-active state into two sub-states: *data reception* and *idle*, as shown in Fig. 4(b). In particular, a UE consumes 500 mW and 255.5 mW per TTI when it is in the data-reception and idle states, respectively. The UE can switch to either the deep-sleep or light-sleep states immediately (i.e., with 0 TTI). When the UE stays in the deep-sleep state, it spends no energy. However, it takes one TTI and 22 mW for the UE to change to the light-sleep state. On the other hand, the UE consumes 11 mW per TTI when it is in the light-sleep state. Besides, the UE will spend one TTI and 39 mW to switch to the idle state.

In fact, it has been pointed out in [37] that both 3-state and Nokia models behave similarly in essence. Moreover, the Nokia model explicitly indicates the amount of time and energy consumption for a UE to switch between each state. Therefore, we choose the Nokia model in our simulations. However, since JIPM does not consider the light-sleep state (i.e., short DRX cycle), a UE directly changes from the deep-sleep state to the idle state, which spends 39 mW to activate its transceiver.

5.2 Simulation Settings

Table 3 gives simulation parameters of eNBs in LTE-Sim, where ζ is the distance between the eNB and a UE, which is measured in kilometers and meters for macrocells and picocells, respectively. Since the LTE-Sim program does not support the ABS method, we do two modifications. First, as LTE-Sim applies the small-cell scenario of femtocells, we increase the transmitted power of a femtocell eNB to 30 dBm to imitate a picocell

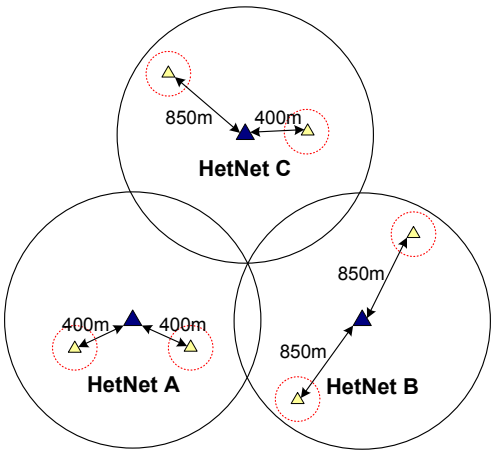


Fig. 5: Network topology in the simulations.

eNB. Second, we let the macrocell eNB broadcast a 30-byte packet using only 1/5 of its transmitted power to simulate the transmission of low-powered control signals in an ABS.

Fig. 5 gives our network topology, which consists of three basic HetNets, where a basic HetNet comprises one macrocell and two picocells. Each basic HetNet has a similar number of UEs but we use different distributions of UEs in these HetNets. Specifically, In HetNet A, the macrocell contains 5/6 of UEs, and each picocell contains 1/12 of UEs. In HetNet B, each cell contains 1/3 of UEs. In HetNet C, the macrocell contains 1/2 of UEs, and each picocell contains 1/4 of UEs. UEs roam in the network following the random-waypoint mobility model [39], where the moving velocity of a UE is set to 3 km/hr to simulate the human walking speed.

Three types of flows are considered: 1) 8.4 kbps VoIP flow (QCI = 1, GBR), 2) 242 kbps video flow (QCI = 2, GBR), and 3) 12 kbps constant-bit-rate flow (QCI = 6, non-GBR). The delay budget of GBR flows is 100 ms. A UE randomly generates two types of flows. We use the *modified largest weighted delay first (M-LWDF)* scheme [40] to allocate PRBs, which benefits GBR flows by considering their packet delays.

Besides, four DRX-configuration schemes are used to compare with JIPM:

- *No DRX cycle (NDC)*: UEs never go to sleep. We use NDC to study the effect without using DRX.
- *Short-length cycle (SLC)*: We set P_i^{SC} to 20 ms and T_i^{ON} to 2 ms to observe the effect of using short DRX cycles.
- *Long-length cycle (LLC)*: We set P_i^{LC} to 100 ms and T_i^{ON} to 10 ms to investigate the effect of using long DRX cycles.
- *Dynamic scheduling with extensible allocation and dispersed offsets (DXD)* [24]: As discussed in Section 3.2, it sets P_i^{SC} as the minimum delay budget of flows and T_i^{ON} as the transmission time based on CQI.

Since these schemes do not adopt the ABS method, we measure their performance with different ABS ratios by setting δ to 0.1, 0.5, and 0.9.

5.3 Energy Efficiency

We first measure the amount of energy efficiency, which is defined by the ratio of network throughput to energy consumption of UEs. In general, higher energy efficiency implies that UEs spend less energy to achieve higher throughput (i.e., they can better utilize their energy to get data). Fig. 6 gives the simulation result by increasing the number of UEs from 90 to 360. NDC always has the lowest energy efficiency, as it does not allow UEs sleeping to save energy (even if they have nothing to receive). Both SLC and LLC use fixed DRX parameters, so they also result in lower energy efficiency, as they cannot change the sleeping and awake time of UEs. Since SLC uses short DRX cycles to let UEs receive more data, it can overtake LLC with more UEs.

Both DXD and JIPM adjust T_i^{ON} and T_i^{OFF} for each UE based on the network condition, so they achieve much higher energy efficiency than others. When the number of UEs grows, their energy efficiency decreases, as network throughput is saturated but more UEs spend energy on getting data. For DXD, a larger ABS ratio δ results in lower energy efficiency. The reason is that macrocell throughput greatly reduces in DXD when δ grows. JIPM integrates the ABS and DRX methods, so it can adaptively co-adjust δ and DRX parameters. Thus, JIPM always has higher energy efficiency than DXD, and the gap enlarges when δ increases.

5.4 Network Throughput

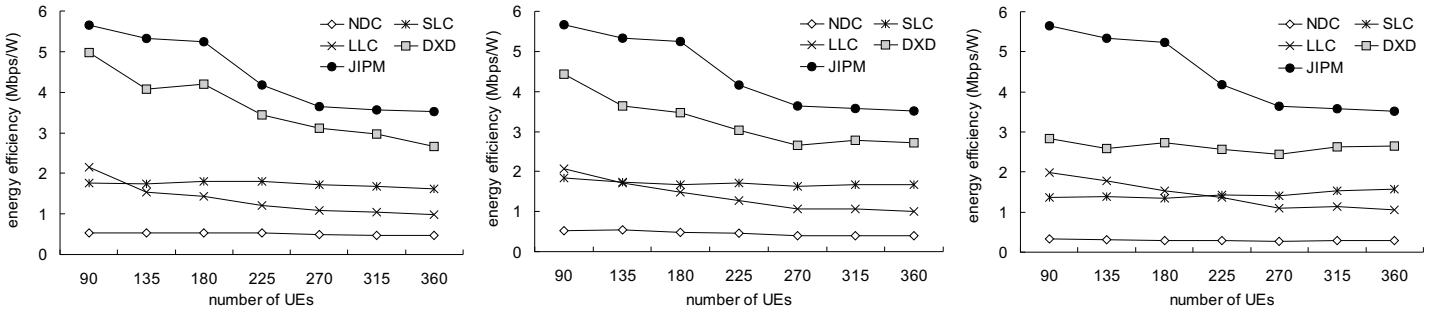
We then study network throughput in Fig. 7. In general, the more the UEs, the higher the throughput. Except for JIPM, network throughput greatly decreases in all schemes when δ grows, because there are more ABSs to let macrocell eNBs stop sending data. From Fig. 7, NDC has higher throughput, followed by DXD, SLC, and LLC. The reason is that NDC makes UEs always active to get data while DXD adjusts DRX parameters based on CQI, so they achieve higher throughput than SLC and LLC. By jointly managing cell interference and UE power, JIPM has the highest throughput among all schemes in most cases.

We also evaluate the percentage of macrocell throughput to total throughput by each scheme, as Fig. 8 shows. For NDC, SLC, LLC, and DXD, the percentage obviously goes down as δ increases, since macrocell UEs get fewer data with a growing number of ABSs. When $\delta = 0.9$, both SLC and LLC have little macrocell throughput. In NDC and DXD, macrocells contribute less than 10% of total throughput with more than 180 UEs. The result shows that these schemes starve macrocell UEs with a large ABS ratio, even if around 55% of UEs are served by macrocells. On the contrary, JIPM uses Eq. (17) to balance cell loads, so its percentage can keep higher than 40%.

5.5 Energy Consumption of UEs

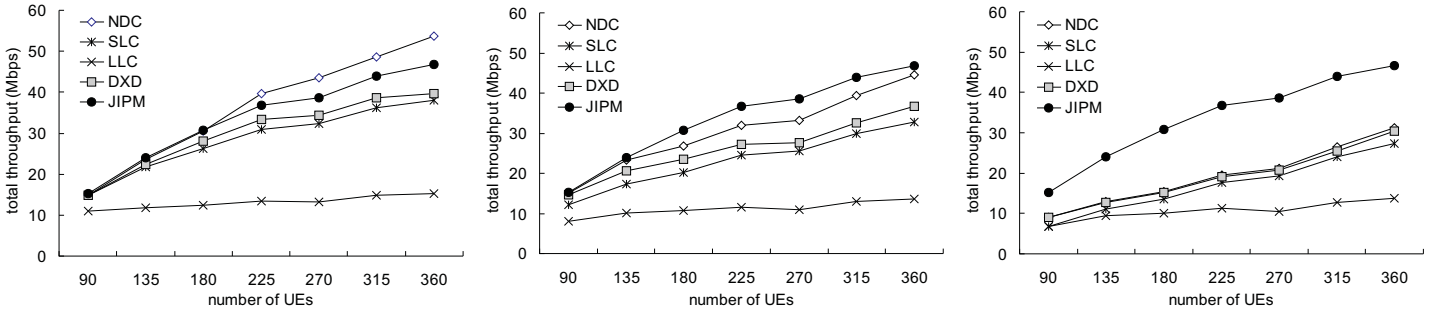
Fig. 9 gives the amount of energy spent by UEs in different schemes. Obviously, NDC lets UEs consume much more energy than other schemes that use the DRX method, which shows the superiority of DRX in energy conservation. On the other hand, SLC makes UEs spend more energy than LLC, DXD, and JIPM. The reason is that SLC uses fixed short DRX cycles, so UEs have to wake up (and spend energy accordingly) more frequently. This experiment verifies that JIPM can achieve lower energy consumption of UEs as well as both LLC and DXD.

Recall that we give an intuitive method to deploy the PRBs of UEs based on their sequence in Section 4.2.4 (referring to Fig. 3(a)). We also compare the amount of energy consumption by this intuitive method (denoted by "JIPM-SEQ") and the PDMAS method (denoted by "JIPM-PDMAS"). Fig. 10 presents the experimental result. Obviously, when there are



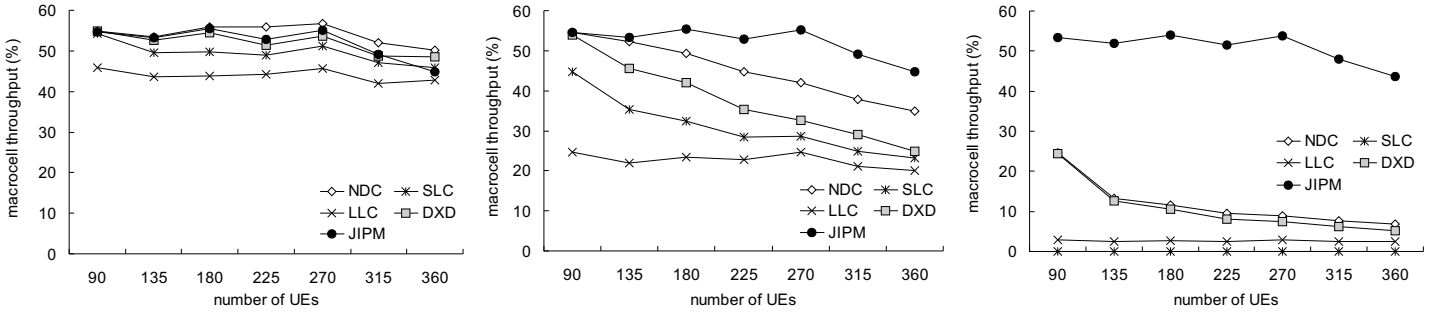
(a) $\delta = 0.1$ (b) $\delta = 0.5$ (c) $\delta = 0.9$

Fig. 6: Comparison on the amount of energy efficiency.



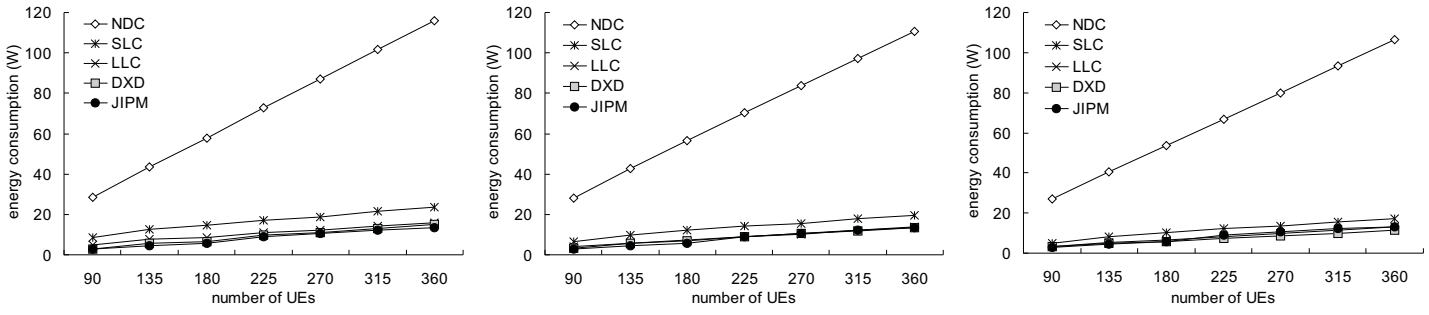
(a) $\delta = 0.1$ (b) $\delta = 0.5$ (c) $\delta = 0.9$

Fig. 7: Comparison on total throughput in the network.



(a) $\delta = 0.1$ (b) $\delta = 0.5$ (c) $\delta = 0.9$

Fig. 8: Comparison on the percentage of macrocell throughput to total throughput.



(a) $\delta = 0.1$ (b) $\delta = 0.5$ (c) $\delta = 0.9$

Fig. 9: Comparison on the amount of energy consumed by UEs.

more UEs, the PDMAS method can save more energy than the intuitive method. The reason is that the PDMAS method can find out more UEs such that their sum of allocated PRBs is dividable by $\hat{N}(B_j)$ and thus shrink their aware time. In particular, the PDMAS method further saves 1.71% to 4.39% energy of UEs than the intuitive method, when the number of UEs increases from 90 to 360.

5.6 GBR Packet Dropping

Fig. 11 shows the dropping ratio of GBR packets. Intuitively, the dropping ratio can reduce by increasing throughput. From Fig. 7, NDC has higher throughput, followed by DXD, SLC, and LLC. Thus, LLC results in the highest dropping ratio, followed by SLC, DXD, and NDC. Besides, these schemes have higher dropping ratios with a larger ABS ratio. Comparing

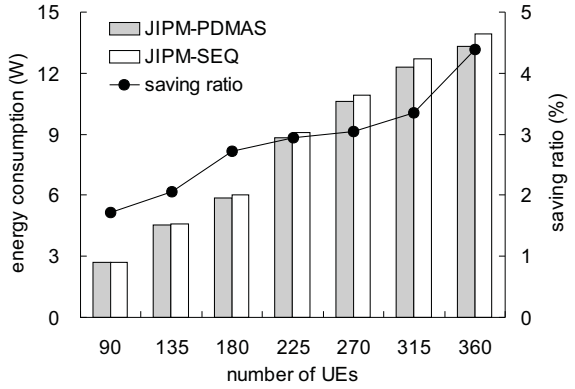


Fig. 10: Comparison on the amount of energy consumption by different PRB deployment methods in JIPM.

with them, JIPM gives a high priority to GBR flows in terms of resource allocation, so it always keeps the lowest dropping ratio. This experiment verifies that JIPM can better support QoS for GBR applications.

5.7 Effect of UE Distribution

Finally, we investigate how different UE distributions affect performance of NDC, SLC, LLC, DXD, and JIPM, where δ is set to 0.5. Fig. 12 gives the amount of energy efficiency in the three basic HetNets, and Fig. 13 shows their dropping ratios of GBR packets. Recall that there are 5/6, 1/3, and 1/2 of UEs served by the macrocell in HetNets A, B, and C, respectively. Since the ABS method benefits picocells by reducing the transmission opportunity of the macrocell, all schemes result in the worst performance in HetNet A. From Fig. 12, JIPM always has the best energy efficiency in each HetNet. Moreover, it results in a lower dropping ratio than others in Fig. 13. These results demonstrate that JIPM performs well under different distributions of UEs, which verifies the effectiveness of its joint management of interference and power.

6 Conclusion and Future Work

LTE-A uses the ABS and DRX methods to reduce signal interference and save energy of UEs, respectively. Existing studies aim to either find the ABS ratio or configure the DRX parameters. None of them considers integrating these two methods, even though they share some similarities. Therefore, this paper develops a four-stage JIPM mechanism to co-adjust the parameters of ABS and DRX, so as to improve LTE-A performance. JIPM finds CQIs of each UE with or without ABS, and estimates the amount of resources required by the UE. It computes the ABS ratio based on the principles of raising picocell GBR throughput and avoiding starving macrocell UEs. Then, JIPM decides both T_i^{ON} and T_i^{OFF} values of a UE to shorten its wake-up time for energy conservation. Through simulations by LTE-Sim, we show that JIPM greatly improves energy efficiency, which means that it allows UEs receiving more data by spending less energy. Comparing with other schemes, JIPM also results in a much lower dropping ratio of GBR packets, which verifies that it can better support QoS for real-time traffics.

However, JIPM addresses only DRX long cycles for the purpose of energy saving. In the future work, we will consider adjusting the parameters of short cycles (e.g., P_i^{SC}) to provide fine-tuning of the DRX method. Besides, JIPM first decides the ABS ratio followed by DRX parameters. It deserves further investigation on how to feed back the result of DRX configuration to the decision of the ABS ratio, so as to achieve tighter combination of both ABS and DRX methods.

References

[1] C. Saha, M. Afshang, and H.S. Dhillon, "Enriched K-tier HetNet model to enable the analysis of user-centric small cell deployments," *IEEE Trans. Wireless Comm.*, vol. 16, no. 3, pp. 1593–1608, 2017.

[2] Y.C. Wang and K.C. Chien, "EPS: energy-efficient pricing and resource scheduling in LTE-A heterogeneous networks," *IEEE Trans. Vehicular Technology*, vol. 67, no. 9, pp. 8832–8845, 2018.

[3] A. Gupta and R.K. Jha, "A survey of 5G network: architecture and emerging technologies," *IEEE Access*, vol. 3, pp. 1206–1232, 2015.

[4] C. Kosta, B. Hunt, A.U. Quddus, and R. Tafazolli, "On interference avoidance through inter-cell interference coordination (ICIC) based on OFDMA mobile systems," *IEEE Comm. Surveys & Tutorials*, vol. 15, no. 3, pp. 973–995, 2013.

[5] Y.C. Wang and S.T. Chen, "Adaptive configuration of time-domain eICIC to support multimedia communications in LTE-A heterogeneous networks," *Proc. IEEE Int'l Symp. a World of Wireless Mobile and Multimedia Networks*, 2017, pp. 1–6.

[6] C.S. Bontu and E. Illidge, "DRX mechanism for power saving in LTE," *IEEE Comm. Magazine*, vol. 47, no. 6, pp. 48–55, 2009.

[7] Y.C. Wang and C.A. Chuang, "Efficient eNB deployment strategy for heterogeneous cells in 4G LTE systems," *Computer Networks*, vol. 79, pp. 297–312, 2015.

[8] J. Pang, J. Wang, D. Wang, G. Shen, Q. Jiang, and J. Liu, "Optimized time-domain resource partitioning for enhanced inter-cell interference coordination in heterogeneous networks," *Proc. IEEE Wireless Comm. and Networking Conf.*, 2012, pp. 1613–1617.

[9] N. Trabelsi, L. Roullet, and A. Feki, "A generic framework for dynamic eICIC optimization in LTE heterogeneous networks," *Proc. IEEE Vehicular Technology Conf.*, 2014, pp. 1–6.

[10] J.A. Ayala-Romero, J.J. Alcaraz, J. Vales-Alonso, and E. Egea-Lopez, "Online optimization of interference coordination parameters in small cell networks," *IEEE Trans. Wireless Comm.*, vol. 16, no. 10, pp. 6635–6647, 2017.

[11] M. Al-Rawi, J. Huschke, and M. Sedra, "Dynamic protected-subframe density configuration in LTE heterogeneous networks," *Proc. IEEE Int'l Conf. Computer Comm. and Networks*, 2012, pp. 1–6.

[12] S. Vasudevan, R.N. Pupala, and K. Sivanesan, "Dynamic eICIC: a proactive strategy for improving spectral efficiencies of heterogeneous LTE cellular networks by leveraging user mobility and traffic dynamics," *IEEE Trans. Wireless Comm.*, vol. 12, no. 10, pp. 4956–4969, 2013.

[13] S.H. Lu, W.P. Lai, and L.C. Wang, "Time domain coordination for inter-cell interference reduction in LTE hierarchical cellular systems," *Proc. IEEE Int'l Conf. Heterogeneous Networking for Quality, Reliability, Security and Robustness*, 2014, pp. 51–55.

[14] A. Daeinabi, K. Sandrasegaran, and P. Ghosal, "An enhanced intercell interference coordination scheme using fuzzy logic controller in LTE-advanced heterogeneous networks," *Proc. IEEE Int'l Symp. Wireless Personal Multimedia Comm.*, 2014, pp. 520–525.

[15] H. Zhou, Y. Ji, X. Wang, and S. Yamada, "eICIC configuration algorithm with service scalability in heterogeneous cellular networks," *IEEE/ACM Trans. Networking*, vol. 25, no. 1, pp. 520–535, 2017.

[16] A. Daeinabi, K. Sandrasegaran, and S. Barua, "A dynamic almost blank subframe scheme for video streaming traffic model in heterogeneous networks," *Proc. IEEE Int'l Conf. Electrical Engineering/Electronics, Computer, Telecomm. and Information Technology*, 2015, pp. 1–6.

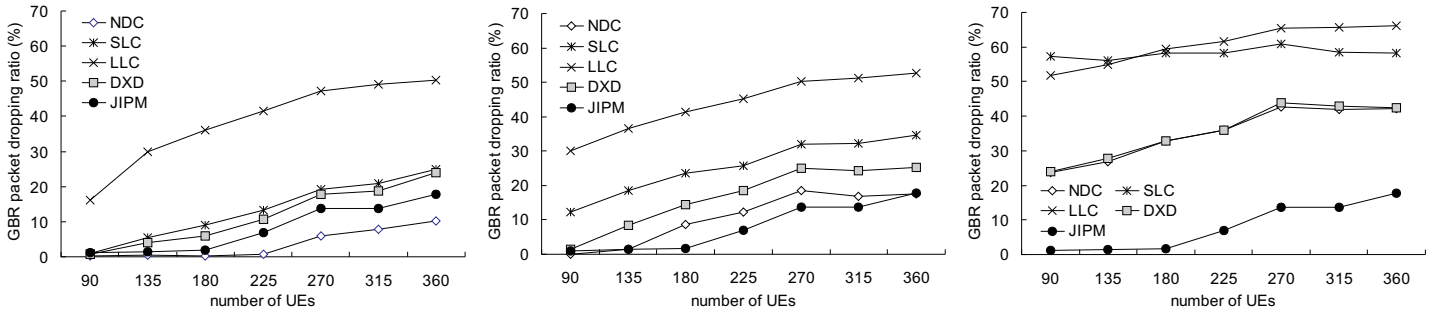
[17] Y.C. Wang and S.T. Chen, "Delay-aware ABS adjustment to support QoS for real-time traffic in LTE-A HetNet," *IEEE Wireless Comm. Letters*, vol. 6, no. 5, pp. 590–593, 2017.

[18] L. Liu, X. She, and L. Chen, "Multi-user and channel dependent scheduling based adaptive power saving for LTE and beyond system," *Proc. Asia-Pacific Conf. Comm.*, 2010, pp. 118–122.

[19] S. Gao, H. Tian, J. Zhu, and L. Chen, "A more power-efficient adaptive discontinuous reception mechanism in LTE," *Proc. IEEE Vehicular Technology Conf.*, 2011, pp. 1–5.

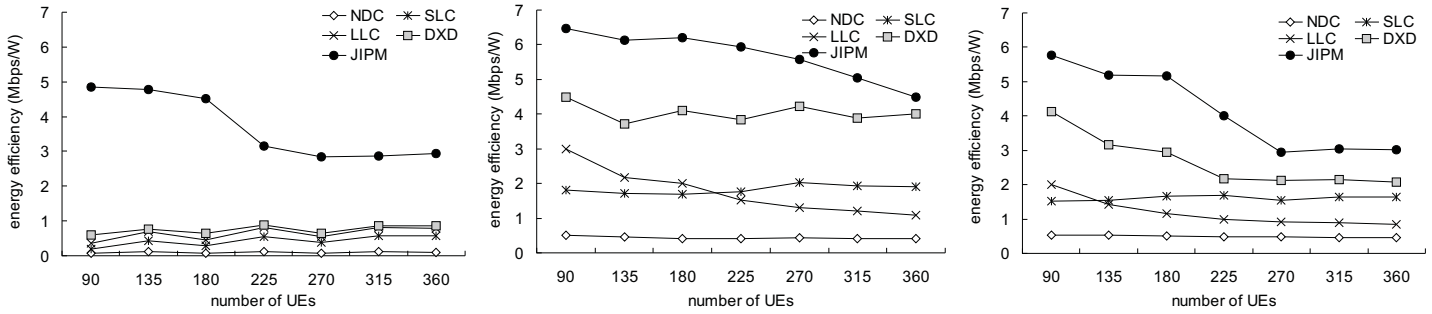
[20] S.C. Jha, A.T. Koc, and R. Vannithamby, "Optimization of discontinuous reception (DRX) for mobile Internet applications over LTE," *Proc. IEEE Vehicular Technology Conf.*, 2012, pp. 1–5.

[21] Y.P. Yu and K.T. Feng, "Traffic-based DRX cycles adjustment scheme for 3GPP LTE systems," *Proc. IEEE Vehicular Technology Conf.*, 2012, pp. 1–5.



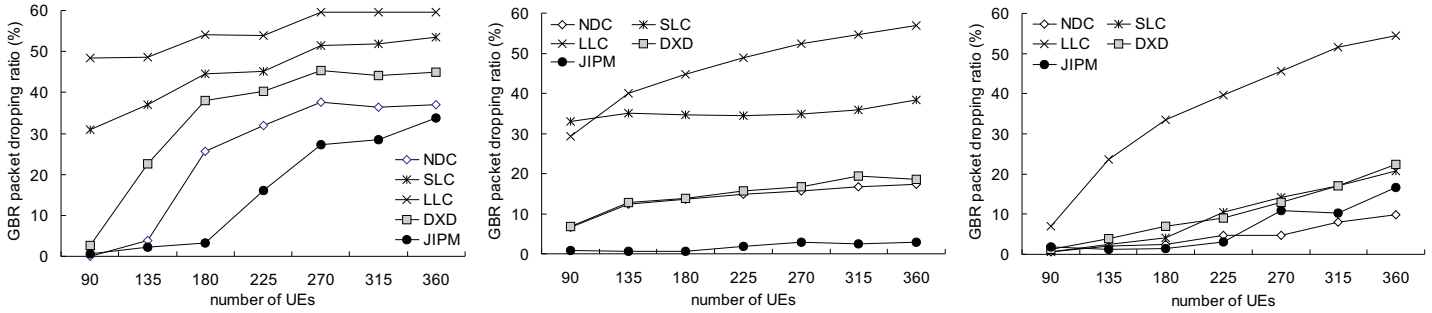
(a) $\delta = 0.1$ (b) $\delta = 0.5$ (c) $\delta = 0.9$

Fig. 11: Comparison on the dropping ratio of GBR packets.



(a) HetNet A (b) HetNet B (c) HetNet C

Fig. 12: Comparison on the amount of energy efficiency in each basic HetNet.



(a) HetNet A (b) HetNet B (c) HetNet C

Fig. 13: Comparison on the dropping ratio of GBR packets in each basic HetNet.

[22] S. Varma, K.M. Sivalingam, L.P. Tung, and Y.D. Lin, "Dynamic DRX algorithms for reduced energy consumption and delay in LTE networks," *Proc. IFIP Wireless Days*, 2014, pp. 1–8.

[23] J.M. Liang, J.J. Chen, H.H. Cheng, and Y.C. Tseng, "An energy-efficient sleep scheduling with QoS consideration in 3GPP LTE-advanced networks for Internet of things," *IEEE J. Emerging and Selected Topics in Circuits and Systems*, vol. 3, no. 1, pp. 13–22, 2013.

[24] L.P. Tung, Y.D. Lin, Y.H. Ku, Y.C. Lai, and K.M. Sivalingam, "Reducing power consumption in LTE data scheduling with the constraints of channel condition and QoS," *Computer Networks*, vol. 75, pp. 149–159, 2014.

[25] J.M. Liang, J.J. Chen, P.C. Hsieh, and Y.C. Tseng, "Two-phase multicast DRX scheduling for 3GPP LTE-advanced networks," *IEEE Trans. Mobile Computing*, vol. 15, no. 7, pp. 1839–1849, 2016.

[26] M.S. Mushtaq, A. Mellouk, B. Augustin, and S. Fowler, "QoE power-efficient multimedia delivery method for LTE-A," *IEEE Systems J.*, vol. 10, no. 2, pp. 749–760, 2016.

[27] R. Giuliano and F. Mazzenga, "Exponential effective SINR approximations for OFDM/OFDMA-based cellular system planning," *IEEE Trans. Wireless Comm.*, vol. 8, no. 9, pp. 4434–4439, 2009.

[28] C. Mehlfuhrer, M. Wrulich, J.C. Ikuno, D. Bosanska, and M. Rupp, "Simulating the long term evolution physical layer," *Proc. European Signal Processing Conf.*, 2009, pp. 1471–1478.

[29] ETSI, "Evolved universal terrestrial radio access (E-UTRA); physical layer procedures (release 14)," 3GPP TS 36.213 V14.1.0, 2016.

[30] F. Capozzi, G. Piro, L.A. Grieco, G. Boggia, and P. Camarda, "Downlink packet scheduling in LTE cellular networks: key design issues and a survey," *IEEE Comm. Surveys & Tutorials*, vol. 15, no. 2, pp. 678–700, 2013.

[31] Y.C. Wang and S.Y. Hsieh, "Service-differentiated downlink flow scheduling to support QoS in long term evolution," *Computer Networks*, vol. 94, pp. 344–359, 2016.

[32] Y.C. Wang and D.R. Jhong, "Efficient allocation of LTE downlink spectral resource to improve fairness and throughput," *Int'l J. Comm. Systems*, vol. 30, no. 14, pp. 1–13, 2017.

[33] H. Kellerer, R. Mansini, and M.G. Speranza, "Two linear approximation algorithms for the subset-sum problem," *European J. Operational Research*, vol. 120, no. 2, pp. 289–296, 2000.

[34] G. Piro, L.A. Grieco, G. Boggia, F. Capozzi, and P. Camarda, "Simulating LTE cellular systems: an open-source framework," *IEEE Trans. Vehicular Technology*, vol. 60, no. 2, pp. 498–513, 2011.

[35] S. Fowler, R.S. Bhamber, and A. Mellouk, "Analysis of adjustable and fixed DRX mechanism for power saving in LTE/LTE-advanced," *Proc. IEEE Int'l Conf. Comm.*, 2012, pp. 1964–1969.

[36] Nokia, "DRX parameters in LTE," 3GPP TSG-RAN WG2 Meeting R2-071285, Tech. Rep., 2007.

[37] S. Fowler, G. Baravdish, and D. Yuan, "Numerical analysis of an industrial power saving mechanism in LTE," *Proc. IEEE Int'l Conf. Comm.*, 2014, pp. 1748–1753.

[38] Y.C. Wang and S. Lee, "Small-cell planning in LTE HetNet to improve energy efficiency," *Int'l J. Comm. Systems*, vol. 31, no. 5, pp. 1–18, 2018.

[39] W.H. Yang, Y.C. Wang, Y.C. Tseng, and B.S.P. Lin, "Energy-efficient network selection with mobility pattern awareness in an integrated

WiMAX and WiFi network," *Int'l J. Comm. Systems*, vol. 23, no. 2, pp. 213–230, 2010.

[40] M. Andrews, K. Kumaran, K. Ramanan, A. Stolyar, P. Whiting, and R. Vijayakumar, "Providing quality of service over a shared wireless link," *IEEE Comm. Magazine*, vol. 39, no. 2, pp. 150–154, 2001.