Machine Learning Enhanced Channel Selection for Unlicensed LTE

Matthew Tonnemacher*, Chance Tarver†, Joseph Cavallaro‡, and Joseph Camp§

*Standards and Mobility Innovation (SMI) Lab, Samsung Research America, Plano, TX, USA

†‡Department of Electrical and Computer Engineering, Rice University, Houston, TX, USA

*§Department of Electrical and Computer Engineering, Southern Methodist University, Dallas, TX, USA

Abstract—We propose a mechanism for unlicensed LTE channel selection that not only takes into account interference to and from Wi-Fi access points but also considers other LTE operators in the unlicensed band. By collecting channel utilization statistics and sharing this information periodically with other unlicensed LTE eNBs, each eNB can improve their channel selection given their limited knowledge of the full topology. While comparing our algorithm to existing solutions, we find that the similarity between sensed Wi-Fi occupation at neighboring eNBs greatly impacts the performance of channel selection algorithms. To achieve better performance across diverse scenarios, we expand on our statistical channel selection formulation to include reinforcement learning, thereby balancing the shared contextual information with historical performance. We simulate operation in the unlicensed band using our channel selection algorithm and show how Wi-Fi load and inter-cell interference estimation can jointly be used to select transmission channels for all small cells in the network. Our approaches lead to an increase in user-perceived throughput and spectral efficiency across the entire band when compared to the greedy channel selection.

Index Terms—LTE-U, LAA, LBT

I. INTRODUCTION

Deployment of LTE technology on unlicensed bands allows new spectrum opportunities for cellular-based mobile broadband. This new spectrum helps to improve downlink data rates, increase network capacity, and, being free for use, allows for new use cases such as neutral host. Moreover, the low-power small/femto cells promise low spatial radio footprints that enable multiplicative capacity gains via frequency reuse. However, being unlicensed, it is possible to face significant interference from incumbent Wi-Fi devices and other unlicensed LTE operators.

Many other works have studied Wi-Fi coexistence with specific implementations of unlicensed LTE (generically referred to as uLTE hereafter), such as LTE-U and its sister technology, Licensed Assisted Access (LAA), and found that LTE can coexist with Wi-Fi better than Wi-Fi can coexist with itself [1–3]. As operator deployment on the unlicensed band ramps up [4, 5], it is necessary to turn our attention to studying coexistence between LTE cells operating in the unlicensed band, which is largely neglected by past works.
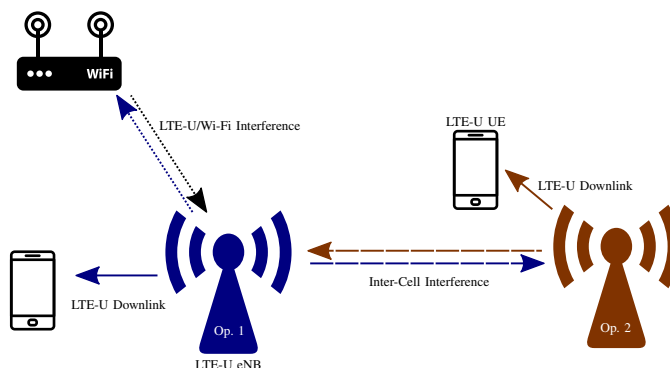
Figure 1. Unlicensed band deployment scenario. In practice, there may be many incumbent Wi-Fi devices, new uLTE eNBs from multiple operators (Ops), and their clients all possibly causing interference with each other. Some sort of interference/collision avoidance mechanism will be necessary to maintain a minimum QoS on the band.

Figure 1 shows a simple illustration of the challenges we face on unlicensed spectrum. In dense deployments, there may be many uLTE eNBs from many operators, uLTE user equipments (UEs) communicating with uLTE Evolved Node Bs (eNBs), Wi-Fi access points (APs), and Wi-Fi clients operating on the same or adjacent channels potentially causing interference.

There are two obvious ways to attempt to improve spectrum sharing: improve sharing in the time-domain via improved channel contention mechanisms, or improve sharing in the frequency domain via improved channel selection. There has been no shortage of papers on the issue of improving time-domain sharing via contention and Listen Before Talk (LBT) schemes [6]. However, the LBT mechanism is difficult to optimize for since there are many competing regulatory requirements. Alternatively, channel selection is not as burdened by such issues, and improvements to the channel selection procedure can yield concrete gains, agnostic to any specific contention protocol.

We make the following observation and seek to exploit it in this work. *If the full unlicensed band channel utilization at each small cell can be estimated and shared with other small cells, a statistical model can be built for each cell selecting the best channel to transmit such that expected network capacity is maximized.* This has the effect of not only improving uLTE/Wi-Fi coexistence but also improving uLTE

inter-operator coexistence. Therefore, in this paper, we seek to improve channel selection by sharing and leveraging statistical modeling of channel utilization over time between neighboring unlicensed eNBs. To do so, we first present a framework that allows uLTE small cells to collect and share their estimated local channel utilizations with neighboring cells. Next, we propose a novel reinforcement algorithm in which each small cell can leverage channel utilization information provided by neighboring cells to select the best operating channel. Finally, we implement, simulate, and compare the performance of our proposed algorithm against the few existing state-of-the-art schemes. We find that by leveraging shared information and reinforcement learning techniques for proper channel selection, aggregate eNB performance is improved for both random scenarios and ones with a high degree of spectral congestion from Wi-Fi devices by at least 10% over existing solutions. We qualify these results by comparing channel selection algorithm performance under specific network conditions to highlight algorithm strengths and weaknesses.

The rest of the paper is organized as follows. We first review related works in Section II. In Section III, we present a mathematical model of the problem and then develop our channel selection algorithm. We simulate and compare our channel selection algorithm to existing and other proposed solutions in Section IV. Finally, we conclude this work in Section V.

II. Related Work

Many works in the literature are concerned with the general topic of LTE on unlicensed spectrum [1–10, 13–19]. However, comparatively few are concerned with channel selection or LTE to LTE coexistence on unlicensed spectrum. In this section, we discuss other works that investigate this problem.

In [10] and [16], the authors focus on channel selection and propose a Q-learning, reinforcement algorithm with [10] originally formulating the algorithm and [16] expanding on it for non-stationary environments. However, their Q-learning is based entirely on past throughput that each small cell achieves on the channel, neglecting the utilization of the channel. Moreover, throughput is realistically a function of more than interference levels, i.e., traffic rates, modulation and coding scheme (MCS), number of users, etc. This could cause the Q-Learning to take unnecessary actions depending on the instantaneous status of the other settings, especially in a dynamic environment. It will also take a long time to explore the entire Q-table and will not be able to adapt to rapid fluctuations in utilizations across the channels. Moreover, the authors do not place Wi-Fi devices in any of their simulations. The authors later also considered a game theory based approach and found it to converge faster than their Q-learning method [15]. One notable strength in all of their work is that their methods are entirely decentralized with each small cell making decisions based on their own throughput.

An alternative approach is taken in [17] where the authors instead consider a joint optimization of the channel selection and the frame scheduling to improve coexistence in the frequency and time domains. Fairness to Wi-Fi devices is included in the optimization as a constraint while maximizing LTE throughput. However, this work relies on a complicated cloud radio access network (RAN) scenario, which would be unlikely to account for multiple operators.

Deep reinforcement learning was used in [18]. This work is also entirely distributed. Although their results are promising, the training data may be prohibitive as multiple days of Wi-Fi utilizations and neighboring eNBs traffic loads are required. Moreover, online training to continuously update traffic models may be too computationally complex to do for each small cell in a network. The Wi-Fi loads dataset used in this work for training and evaluating their method is from a 2003 dataset that polls for AP activity every five minutes. It is unclear how their method would perform in modern and likely less predictable traffic patterns where there are many more Wi-Fi APs and clients in a region.

III. Channel Selection Algorithm Overview

In this section, we outline various algorithms that could be used for channel selection strategies. However, before doing so, we first provide details relating to our system model.

We consider a field of operation in which there are three types of entities deployed: eNBs operating in the unlicensed band, Wi-Fi APs, and UE devices connected to the eNBs. We lump Wi-Fi device behavior into that of the Wi-Fi AP, as we are only looking at the downlink channel and are not directly considering Wi-Fi performance in this paper, as it is a well-studied topic. Most unlicensed LTE protocols require or at least have some form of LBT available. In our formulation, we consider an abstracted form of LBT between cells and Wi-Fi APs, in which transmission opportunities are slotted. In each slot, eNB transmission requires a clear channel assessment (CCA) be performed to determine whether or not the medium is idle. Since this work focuses primarily on inter-cell coexistence, we do not specifically model Wi-Fi MAC layer operation, instead assuming that Wi-Fi APs capture the channel according to Poisson arrival process with independent uniformly distributed random access duration bounded by the CSMA protocol. The slots selected are assumed to be already captured by the Wi-Fi in simulation without simulating specific Wi-Fi MAC behavior. Each eNB only contends for the channel in slots where Wi-Fi traffic is absent; channel access for the slot is then randomized between all contending eNBs for each slot. By modeling our setup in this way, we can abstract away from specific protocol implementations (allowing for this work to extend towards future spectrum sharing protocol design) and reduce overall variability between simulations. As a consequence of this, the mechanisms and performance trade-offs involved with implementation of each algorithm is not evaluated in this work, as such an evaluation, as such an evaluation would require the study to focus on one specific realization of unlicensed LTE. Moving forward, we define eNBs that exist within contention range of any given eNB as its neighbors. The neighbor set will be used to simplify much of our analysis.
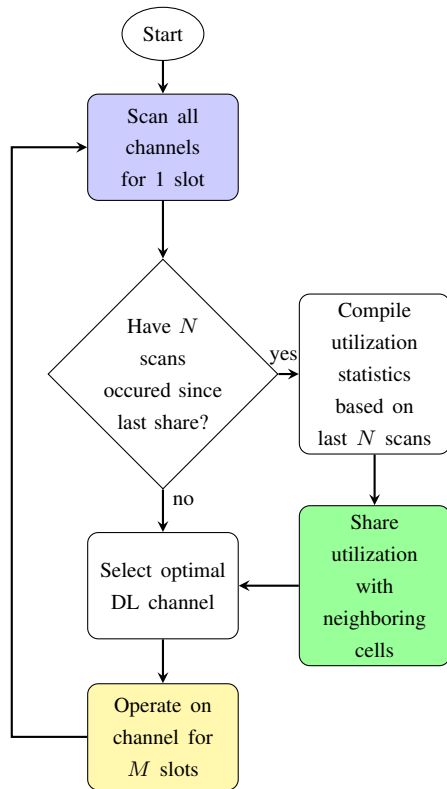
Figure 2. MAC system overview. With a period of every M slots (or 1 epoch), the eNB will operate on a DL channel. The eNB will scan the channel and average utilization statistics over a window of the last N scans.
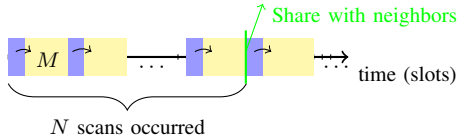

Figure 3. Time domain representation of eNB behavior. An eNB scans all channels for one slot to measure the energy in each channel (shown in blue). Using this info, the eNB chooses a DL channel to operate on for the next M slots (yellow). This process repeats for a total of N scans before the utilization statistics are reported to neighboring cells.

The major objective of this work is to provide an improved methodology for selecting an unlicensed channel for transmission, such that each eNB can the best quality of service for their connected UEs. In order to do so, we modify the fixed-channel operation of LTE and adapt it to our slotted transmission system. The channel selection is done periodically according to the following steps.

1) Each eNB scans the band for 1 full slot to measure the power in each channel.
2) Based on this, it will choose its DL channel to operate on for the next M slots based on the channel selection algorithm implemented.
3) After M slots have passed, the eNB will re-scan the set of channels. We call this duration of M an epoch.
4) After N scan cycles have completed, the eNB will share its current channel utilization statistics with neighboring

eNBs before the channel selection.

A flow chart of the overall system algorithm is shown in Figure 2, and a time domain depiction of basic operation is shown in Figure 3. We note that the 1 slot scanning duration is considered a silent period for the eNB in that it is unable to transmit throughout. Additionally, we assume silent periods are done simultaneously between all eNBs. The primary reason for this is to allow each eNB to capture the Wi-Fi utilization without the addition of inter-cell interference. While this is not strictly necessary due to the ease in which an eNB could differentiate between Wi-Fi and LTE transmissions, it is convenient for notation and simulation implementation. In terms of performance, all presented algorithms (besides the greedy selection) rely only on the measured Wi-Fi interference, as we consider Wi-Fi sources as operating on fixed channels.

Since each scanning period is limited in duration by design, we track the utilization over time via an exponentially weighted moving average as:

$$\mathbf{u_i} \leftarrow \alpha \hat{\mathbf{u}}_\mathbf{i} + (1 - \alpha)\mathbf{u_i}. \tag{1}$$

Here, the moving average for eNB i, $\mathbf{u_i} \in [0,1]^{|\mathcal{K}|}$, is updated by using its past value and most recent measurement $\hat{\mathbf{u}}_\mathbf{i}$, both being vectors representing the utilization on each channel, k, in the set of channels, $\mathcal{K}$. The scalar $\alpha \in [0,1]$ is a weight that balances recent measurements versus past measurements. If, for example, eNB i measures channel k to be fully utilized by Wi-Fi devices in its most recent measurement, then $u_{ik} = 1$. The moving-average formulation strikes a reasonable balance of prioritizing recent measurements while maintaining a long-term history and having low implementation complexity.

A. Greedy channel selection

With a greedy channel selection algorithm, each eNB seeks to optimize its throughput by choosing the channel with the lowest utilization in the previous scan. This is similar to how many uLTE base stations currently operate [7–9]. The idea is to pick the least occupied channel possible, thus minimizing the impact on existing Wi-Fi networks in the area and maximizing channel access for the eNB. This is shown in Equation 2 where k_j is the channel for eNB j, t is the current epoch, and $u_{j,k}$ is the most recent measured utilization by eNB j for channel k.

$$k_j(t+1) = \arg\min_{k \in \mathcal{K}} u_{j,k} \tag{2}$$

The greedy channel selection algorithm has the benefit of being completely decentralized with each eNB making decisions based only on local measurements. However, this simple algorithm may suffer in scenarios where there are many eNBs in close proximity as all eNBs would collide on the channel with the lowest utilization.

B. SOPI — Stochastic Optimization with Partial Information

To improve on the greedy approach, we present two different algorithms that each consider not only local channel utilization measurements but also neighboring measurements.

In both of these algorithms, each eNB locally calculates the expected capacity for the network given its local sensing of the unlicensed channels and neighboring reports of their sensing. For the first algorithm, we use a statistical approach to leverage local and neighboring channel utilization measurements to optimize sum capacity. We consider this an optimization with partial information, as the only information shared from neighbors is the Wi-Fi utilization. This algorithm is performed independently at each eNB. The process of each eNB is to first estimate the expected network capacity for each channel, then select the channel that maximizes this estimated capacity. To properly calculate the expected capacity, we first estimate the joint probability of channel selection between all neighboring eNBs and each combination's respective capacity.

We begin by estimating the probability that an eNB is able to capture the channel for transmission. Each eNB, i, measures the utilization of channel k, u_{ik} given from Eq. 1. For each eNB, we can derive the probability of access, P_a, to channel k as:

$$P_a(i, k) = 1 - u_{ik}. \tag{3}$$

Let eNB i have a set of neighbors, $\mathcal{N}$. We define a subset of neighbors simultaneously operating on the same channel as $J \subseteq \mathcal{N}$. We also define binary indicator vector, $\mathbf{y}$, of length $|J|$ where each element represents the instantaneous channel availability for eNB $j \in J$. All combinations of $J \subseteq \mathcal{N}$ must be iterated through to calculate the expected network capacity. We can then define the probability of eNB i capturing channel k for a slot given the subset of neighbors J also choose k as:

$$P_{\text{capt.}}(i, k, J) = \sum_{\mathbf{y} \in Y} \frac{\prod_{j \in J} \left((1 - P_a(j, k))^{\bar{y}_j} \cdot (P_a(j, k))^{y_j} \right)}{w(\mathbf{y}) + 1} \tag{4}$$

Here, $w(\mathbf{y})$ is the Hamming weight of binary indicator vector $\mathbf{y}$, $\bar{\mathbf{y}}$ is the binary complement of $\mathbf{y}$, and $Y = \mathbb{Z}_2^{|J|}$ is a set containing all realizations of $\mathbf{y}$. This channel capture probability is based on the premise that the channel can only be captured when the same channel neighbors either do not have access to the channel for the slot, or they do have access to the channel, and eNB i wins contention against them. With the channel capture probability found, we calculate the channel capacity for eNB i with contending set J on channel k as:

$$C_i(k, J) = P_a(i, k) P_{\text{capt.}}(i, k, J) C_{i,\max} \tag{5}$$

Here, the probability of transmission is scaled by both the access probability, P_a, for node i from Eq. 3 and the capture probability, $P_{\text{capt.}}$ from Eq. 4, since the channel needs to be available before it can be contended for. We define $C_{i,\max}$ as the unimpeded channel capacity estimated by i based on the average SNR experienced over all connected UEs as $W \log_2(1 + \text{SNR}_i)$. Using a similar method, eNB i estimates the expected capacity of neighboring eNB j by assuming that j's only neighbor is i and that average SNR for j is identical to its own.

$$C_j(k, J) = P_a(j, k) P_{\text{cap}}(j, k, i) C_{i,\max} \tag{6}$$

Now that we have found both the local and neighboring expected capacities for all k and J, we can begin calculating the overall expected network capacity given eNB i selects channel k.

For eNB i, we can estimate the probability that a neighboring eNB $j \in \mathcal{N}$ will have access to channel k by converting the channel availability of each eNB to a probability distribution given as:

$$P_s(j, k) = \frac{1 - u_{jk}}{\sum_{\gamma \in \mathcal{K}} (1 - u_{j\gamma})} \tag{7}$$

While ideally, we would perform a joint optimization over all channels, our model assumes eNB i only has local Wi-Fi utilization of its neighbors and nothing else. Thus, this estimate is subject to error based on discrepancies in actual access probabilities, even if all neighboring eNBs are using identical channel selection strategies. We address this problem in more detail later on.

Next, we derive the probability that multiple neighbors, denoted by the set $J \subseteq \mathcal{N}$, will access the single channel k. To calculate this, we multiply the individual probabilities that a neighbor in J accesses the channel by the probabilities that other neighbors access any other channel. This is given as:

$$P_s(J, k) = \prod_{j \in J} P_s(j, k) \cdot \prod_{j \in \mathcal{N} \setminus J} (1 - P_s(j, k)) \tag{8}$$

The total capacity experienced by the network when eNB i with the subset of neighbors J transmits on channel k is given by:

$$
\begin{aligned}
C_{i,\text{net}}(J, k) =& (1 - u_{ik}) C_i(J, k) \\
& + \sum_{\gamma \in \mathcal{N}} (1 - u_{\gamma k}) \left(x_\gamma C_\gamma(J, k) + \bar{x}_\gamma C_{i,\max} \right) \tag{9}
\end{aligned}
$$

Here, x_γ is an indicator. Namely, 1 if $\gamma \in J$ and 0 for all other cases. The variable $\bar{x}_\gamma$ is an indicator that is the complement to x_γ.

The expected network capacity if eNB i chooses channel k is given by summing the probability that a subset of neighboring eNBs, J, transmit on the channel multiplied by the network capacity in such a scenario over all possible combinations of eNBs in the neighboring set, $\mathcal{N}$, which is given by the power set, $\mathbb{P}(\mathcal{N})$.

$$\hat{C}_{i,\text{net}}(k) = \mathbb{E}\left[C_{i,\text{net}}(J, k) \right] \tag{10}$$

$$= \sum_{J \in \mathbb{P}(\mathcal{N})} P_s(J, k) C_{i,\text{net}}(J, k) \tag{11}$$

The channel chosen by eNB i for epoch $t + 1$ should therefore be selected such that the expected network capacity is maximized according to:

$$k_i(t + 1) = \arg\max_{k \in \mathcal{K}} \hat{C}_{i,\text{net}}(k) \tag{12}$$

The result of this algorithm, then, is to select a channel that maximizes the expected sum capacity in the network. However, because the second-order neighbors (neighbors of

each neighbor to a given eNB i) are unknown, the only second order neighbor we can assume exists is eNB i itself. Without a full neighbor graph, the neighbor channel access probabilities estimated by $P_s(J, k)$ can differ from reality. Unfortunately, in the case where channel utilization statistics are similar across all eNBs for each channel, each eNB will have a high probability of selecting the same channel despite any attempted avoidance due to all eNBs operating under the same channel selection policy. An eNB following the same policy with the same information will necessarily make the same decision in a hard-decision maximization.

In such a situation, the issue of channel selection is a game-theoretical problem. The algorithm performance can be expected to be worse than the greedy selection algorithm under the premise that at least the greedy selection algorithm will lead to mass collisions on the channel with the lowest measured Wi-Fi occupation, whereas the partial information in the SOPI algorithm can drive eNBs to all simultaneously select a busier channel, degrading performance further.

We can partially mitigate the damage in this situation by converting the vector of expected network capacities for each channel, $\hat{C}_{i,\text{net}}(k)$, to a probability distribution, where $P_c(i, k)$ represents the probability of eNB i selecting channel k given by:

$$P_c(i, k) = \frac{\hat{C}_{i,\text{net}}(k)}{\sum_{\gamma \in \mathcal{K}} \left(\hat{C}_{i,\text{net}}(\gamma) \right)} \tag{13}$$

However, such a solution is not ideal as the result-per-game is memory-less and does not converge to a Nash equilibrium over time, resulting in a pseudo-random collision behavior.

C. Romero-Q — Reinforcement learning using throughput from [10]

In [10] the authors adapt reinforcement learning for uLTE channel selection. We detail their algorithm along with a brief introduction to reinforcement learning below.

Reinforcement learning is an iterative interaction between the agent (which hosts the algorithm) and the environment used in many fields. Agents interact with the environment via actions. Each action performed by the agent impacts the environment in some way, causing the environment to emit an observation and reward. The observation and reward are used to refine the decision-making function, or policy, of the agent. Reinforcement learning methods specify how the agent changes its policy as a result of its experience. The agent's goal is to maximize the total amount of reward it receives over the long term.

In [10], a subcategory of reinforcement learning called Q-learning is used. The quality $Q_i(k)$ of eNB i taking the action to select channel k based on the number of available channels, $k \in K$, is defined by:

$$Q_i(k) \leftarrow (1 - \alpha_L) \cdot Q_i(k) + \alpha_L \cdot r_i(k) \tag{14}$$

Here, α_L is the learning rate and $r_i(k)$ is the reward the agent received from the environment for selecting channel k. They

derive the reward is determined based on the achieved throughput seen on channel k and the final decision is probabilistic based on the soft-max policy described in [11]. This decision is given as:

$$P_{Qi}(k) = \frac{\exp\left(\frac{Q_i(k)}{\tau}\right)}{\sum_{\gamma \in \mathcal{K}} \exp\left(\frac{Q_i(\gamma)}{\tau}\right)} \tag{15}$$

D. RLPI — Reinforcement Learning with Partial Information

In order to properly manage situations with similar channel utilizations across neighboring eNBs, we propose a novel reinforcement learning algorithm that combines prior work for LTE-U channel selection from a purely game-theoretical perspective [10] with our analytical model that includes both local and neighboring channel utilization information.

Rather than allocate reward based on achieved throughput as done in [10], we prefer to measure the average SINR over the past epoch and calculate the reward directly using how successful the eNB was at transmitting on the chosen channel. Directly using a notion of contention is a natural choice as throughput is a function of many parameters such as UE placement, MCS, and inter-cell interference. The reward used in this work is given as:

$$r_i(k) = \frac{m_{\text{cap}}(i)}{M}, \tag{16}$$

Here, $m_{\text{cap}}(i)$ is the number of slots successfully transmitted on by eNB i in the previous epoch, and M is the number of slots in each epoch. By using a unit-less reward, we can abstract away many of the dynamic aspects of the system, such as interference, channel fluctuations, UE scheduling, and modulation/coding scheme selection.

We define a metric for similarity, S, based on the Jensen-Shannon divergence between all normalized channel utilizations. This normalization is given by:

$$P_u(j) = \frac{u_{jk}}{\sum_{k' \in \mathcal{K}} u_{jk'}}. \tag{17}$$

Using these normalized channel utilization distributions, we define the channel utilization similarity for eNB i as:

$$S_i = \frac{1 - \sum_{j \in \mathcal{J}} JSD(P_u(i)|P_u(j))^{1/\omega}}{|J|} \tag{18}$$

Here, $JSD(P_u(i)|P_u(j))$ is the Jensen-Shannon divergence between the normalized channel utilization experienced by eNB i and neighbor j. We normalize by the cardinality of J to bound the similarity between $[0, 1]$, where a value of 1 indicates the channel utilization between the eNB i and all of its neighbors is identical, and a value of 0 indicates all neighboring channel utilizations diverge from the locally measured channel utilization. We add an ω exponent as a tunable parameter that functions to scale how aggressively the algorithm relies on the net capacity estimates over the Q-value. Unless specified otherwise, we set $\omega = 2$, which commonly used as the Jensen-Shannon distance [12].

Similar to Romero-Q, RLPI performs channel selection through a soft-decision function based on the soft-max policy [11] combined with the estimated channel capacities. The probability for eNB i to select channel k via the modified soft-max decision is given by:

$$P_{Qi}(k) = \frac{\exp\left(\frac{S_i Q_i \tilde{}(k) + \bar{S}_i \tilde{C}_{i,\text{net}}(k)}{\tau}\right)}{\sum_{\gamma \in \mathcal{K}} \exp\left(\frac{S_i Q_i \tilde{}(\gamma) + \bar{S}_i \tilde{C}_{i,\text{net}}(\gamma)}{\tau}\right)} \qquad (19)$$

Here, $\bar{S}_i$ is the complement to S_i given by $\bar{S}_i = 1 - S_i$, $Q_i\tilde{}(k)$ and $\tilde{C}_{i,\text{net}}(k)$ are the scaled sets of eNB i's Q values and expected network capacities for each channel $k \in K$, such that the maximum value in the set is 1 and other members of the set are scaled relative to it. This scaling to unit values allows the Q values and expected network capacities to be compared on the same relative scale. τ is the temperature, which is a function of the number of epochs experienced by the algorithm as $\tau = \frac{\tau_0}{\log(1+t)}$. The temperature is used as a way to increasingly polarize the selection probabilities the longer the algorithm runs [10]. Consequently, more exploration will occur during the earlier epochs, decreasing as time goes on. The value of τ_0, or the initial temperature, should be set to tune the exploration. Equation 19 aims to strike a balance between past performance represented by the Q value and the expected network capacity for a given channel selection. When the similarity is high, the selection will be primarily based on past behaviors to avoid group-think between all eNBs by using the experience-driven Q-value as the primary selection metric. When the similarity is low, the selection will be primarily based on the relative expected network capacity between selected channels, exploiting the shared information between eNBs.

E. SOFI — Stochastic Optimization with Full Information

To provide a reasonable upper bound for performance, we evaluate a fully-informed algorithm in which the channel selection for each eNB is performed jointly. This algorithm operates as an exhaustive search over all possible combinations assuming the decision is made for all jointly. While this is not possible in a distributed cellular network, it would be possible with centralized base-band processing across all nodes in a cloud-RAN architecture. The SOFI algorithm operates as follows.

Let $\mathbf{k} \in \{1, \ldots, |\mathcal{K}|\}^{|\mathcal{X}|}$ be a vector where each element denotes the channel selected for the eNB with the corresponding index. This vector encapsulates the channel selection for all eNBs in the system, which is denoted by the set $\mathcal{X}$. The idea here is to choose the $\hat{\mathbf{k}}$ that maximizes the net capacity, i.e., we globally assign each eNB a channel instead of having them independently select their channel. This optimization is given by:

$$\mathbf{k} = \arg\max_{\hat{\mathbf{k}}} C_{\text{net}}(\hat{\mathbf{k}}) \qquad (20)$$

To accomplish this, we calculate the resulting expected network capacity for all channel selection vectors $\mathbf{k}$. $P_a(j, k_j)$

is the probability that eNB j can access the channel assigned in the scenario (Eq. 3). $L_j \subseteq \mathcal{X}$ is the set of eNBs on the same channel as eNB j.

$$C_{\text{net}}(\mathbf{k}) = \sum_{j=1}^{|\mathcal{X}|} \frac{P_a(j, k_j)}{1 + |\mathcal{N}_j \cap L_j|} C_{j,\max} \qquad (21)$$

For a channel selection scenario, this equation sums the product of the probability that a node will access the channel, the probability it will win contention, and its max capacity if it were to win contention. The probability of winning contention is given by assuming that all neighboring nodes on the same channel are equally likely to win contention.

This global channel selection is done by explicitly calculating the expected network capacity over all possible vectors $\mathbf{k}$, which may be a large search space in cases when there are many channels and many nodes. This model is a reasonable approximation of an upper bound. However, it is not a true upper bound in the sense that we are unaware of future UE and Wi-Fi traffic.

IV. SYSTEM-LEVEL uLTE SIMULATION

In this section, we compare the relative performance of each algorithm in a system level simulator we developed to model generalized uLTE operation. We first present details of our simulation environment. Using our simulator, we compare the performance of the five presented channel selection algorithms in a randomly-generated topology with fixed parameters. We then evaluate algorithm performance for both high and low average similarity, S, across all deployed eNBs. We further split the high average similarity result into low and high overall Wi-Fi utilization to see the impact of abundant and sparse transmission opportunity availability, respectively, for each algorithm.

A. Simulation Environment

We test the algorithm in MATLAB where we design a simulator that randomly places eNBs and UEs, model interference using the ITU-INH channel model, and generate unique traffic being requested by UEs at each transmission time interval (TTI) according to a fixed traffic rate. TTIs can be equated to slots from our previous formulation, and have a duration of a single LTE subframe, 1 ms. Based on user positions, an appropriate MCS is chosen. If the UE sees severe interference, the cyclic redundancy check (CRC) on their transmission may fail, prompting a re-transmission. We evaluate and update performance metrics upon successful packet reception, which can take several TTIs. In this simulator, we only model the downlink traffic using a 20 MHz, FDD LTE-based signal. As the scope of this work is focused on uLTE performance, the Wi-Fi traffic is modeled as transmissions that occur according to a regular traffic rate without modeling the individual Wi-Fi clients, their detailed MAC behavior, or their throughputs.

The optimization from Equation 12 is performed through a brute-force search. This method is not unreasonable given the relatively small, discrete space that can explicitly be

calculated. However, this space does grow exponentially with the number of available channels and the number of neighbors. In this case, heuristics or advanced optimization techniques can be explored. However, such analysis is outside the scope of this paper.

To quantify performance, we use the user perceived throughput (UPT). The UPT is a metric of the throughput experienced by each individual user after accounting for the total time, including re-transmissions, taken for the complete packet to be received by the user. Using this metric has several beneficial consequences for analysis. UPT is significantly impacted by temporary outages in service. For our application, this amplifies the impact of poor channel selection, as the buffered data for transmission grows. From a UE perspective, UPT is also impacted by fairness and latency. This allows it to be used as a general quality of service metric, something unachievable when comparing throughput alone. We formally define UPT in Equation 22.

$$ UPT = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{P_{total}} \left[\sum_{j=1}^{P_{served}} \frac{M \cdot r_{ij}}{t_{ij}} + \frac{b_i}{t_{serving,i}} \right] \quad (22) $$

Here, N is the number of UEs served by the eNB, and i indexes the UEs. P_{total} is the total number of packets, elaborated by $P_{total} = P_{served} + P_{serving}$, where P_{served} and $P_{serving}$ are the number of packets served and being served, respectively. M is the number of bits per packet, r_{ij} is the ratio of successfully transmitted bits over all bits in the packet to UE i for packet j, and t_{ij} is the time taken to send the same packet. b_i is the number of bits sent to UE i as a partial packet still in flight, and $t_{serving,i}$ is the time spent by the packet.

B. Overall Performance

Using our simulation environment, we first evaluate performance over many different random topologies. By doing so, we can gain an understanding of how each algorithm can be expected to perform in a generalized scenario. For this evaluation, we employ Monte Carlo simulation over many different "drops", or realizations of random simulation parameters. These drop-randomized parameters include eNB, Wi-Fi AP, and UE deployment locations, initial channel assignments, and data traffic realizations.

In Figure 4a, we show an example random realization of the network topology or "drop." Here, the four eNBs and eight Wi-Fi APs are uniformly randomly distributed in a 240×240 meter space. Each eNB has ten connected UEs uniformly distributed in a sixty-meter radius around it.

For this simulation, we set the parameters as shown in Table I and then perform a Monte Carlo simulation. We simulate 100 unique drops with random topologies which are run for 5,000 TTI, or 5 seconds. The relatively short simulation duration functions as a snapshot of performance within the physical network topology, and traffic sources are unlikely to change significantly. Our further assumption is that the performance of these algorithms over this duration is

Table I
PARAMETERS USED FOR OVERALL PERFORMANCE SIMULATION.

| Parameter	Value in Simulation
Number of Channels, $\mathcal{K}$	4
Number of uLTE eNBs, $\mathcal{X}$	4
Number of Wi-Fi APs	8
Scan Period, M	20 TTI
Update Period, N	100 TTI
Deployment Area	240m x 240m
Drops	100
eNB TX Power	18 dBm
Wi-Fi AP TX Power	14 dBm
ED Threshold	-62 dBm

extendable to multiple subsequent durations with incremental time-dependent changes to the topology. We plot the average UPT for each algorithm in Figure 5.

We find, over an extensive series of random topologies, an example of which is shown in Figure 4a, the Greedy algorithm has the worst performance, while RLPI performs the best. It is interesting to note that both of the reinforcement algorithms outperform both SOPI and SOFI algorithms. One explanation for this is the partial and full optimization algorithms in SOPI and SOFI only considers channel selection with regards to channel and Wi-Fi utilization, while remaining oblivious to well known hidden factors such as exposed/hidden terminals between eNBs. Reinforcement learning performs exceptionally well in environments with hidden influences, allowing it to out-perform others in the general case. However, these factors alone may not account for such a discrepancy in performance over a large number of simulation iterations. In the next set of experiments, we isolate the utilization similarity metric, dependence between channel access probabilities, and Wi-Fi traffic load to better explain the underlying causes of performance differences between these algorithms.

C. Utilization Similarity, Dependence, and Load

For our RLPI algorithm, we introduced a similarity metric, S, to balance the Q-value impact and expected network capacity when calculating channel selection probabilities. This similarity metric can be thought of as the relative difference in channel availability an eNB and its neighbors. In other words, a high similarity indicates the Wi-Fi occupancy is roughly the same between an eNB and its neighbors for each channel, and a low similarity indicates the opposite.

However, we can also think of similarity as a metric of how correlated the channel access probabilities between each eNB. For example, if two eNBs share a common Wi-Fi AP nearby on channel k, the access probabilities for the two eNBs on channel k as defined in Equation 3 are not strictly independent, distorting the calculation for capture probabilities in Equation 4. This interaction between instantaneous channel availability correlation and similarity is not straightforward. For example, while a high dependence between channel access probabilities implies a high similarity metric, a high similarity can be realized with completely independent channel access proba-
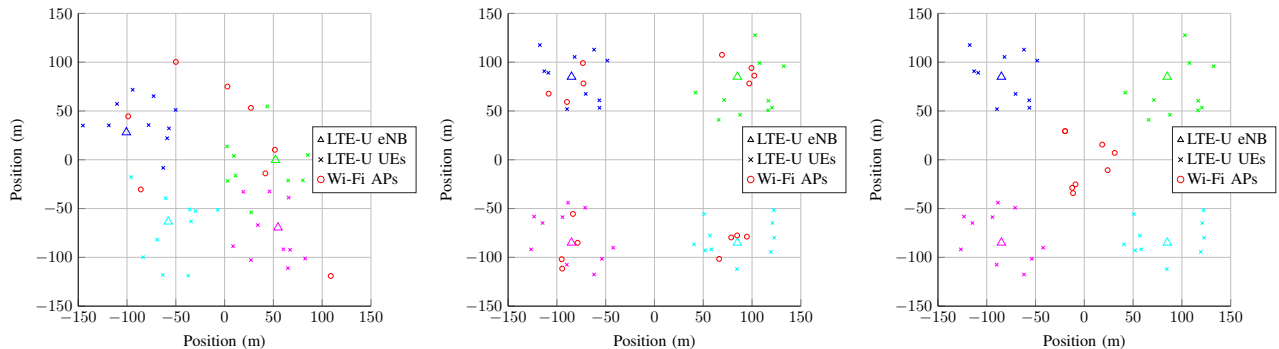
(a) Example realization of the random topology used in Monte Carlo simulation.

(b) Example realization of the fixed topology used for generating a low dependence scenario.

(c) Example realization of the fixed topology used for generating a high dependence scenario.

Figure 4. Example experimental topologies with four separate eNBs each with ten clients and eight Wi-Fi APs. The colors (blue, green, cyan, and magenta) represent an eNB and its clients. The eNBs and the Wi-Fi APs are uniformly randomly distributed in a room that is 240×240 meters. The UEs are distributed randomly in a 60-meter fixed radius around their eNB.
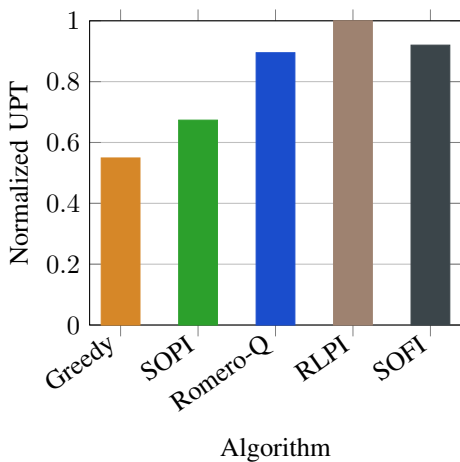

Figure 5. Overall algorithm performance, averaged across 100 random drops over 5,000 TTI to gauge the relative performance of each algorithm in generalized circumstances.

bilities between neighbors. This relationship strictly depends on whether an eNB neighborhood shares common Wi-Fi APs (high dependence) or if they do not share APs, but the utilization happens to be the same (low dependence). Unfortunately, the eNBs and, by extension, the channel selection algorithm they employ are unable to differentiate the access probability dependence and must rely on the similarity metric alone.

Thus, we need to explicitly quantify the impact of similarity on the performance for each algorithm. To do so, we first define an average similarity metric, S_{avg} as the average similarity for a simulation as:

$$S_{avg} = \sum_{i \in \mathcal{X}} \frac{S_i}{|\mathcal{X}|} \qquad (23)$$

For our analysis, we categorize similarities into two extreme scenarios: low average similarity, $S_{avg} < 0.5$, and high average similarity, $S_{avg} > 0.9$. The results for similarity values in-between the high and low can be considered the general

case, which represents the majority of drop outcomes in the Monte Carlo simulation. To elucidate these thresholds, we have designed representative scenarios for each.

a) Low similarity: To properly analyze low similarity, we modify our simulation environment to provide a more deterministic topology for study. We fix the positions of 4 eNBs in a square pattern such that each eNB is considered a neighbor to the two closest eNBs, creating a unique neighbor set for each eNB. Four Wi-Fi APs are distributed near each eNB such that other eNBs are not impacted. This creates completely independent Wi-Fi interference sources for each of the eNBs. An example deployment under this fixed topology constraint is shown in Figure 4b.

To achieve the desired similarity for simulation, we randomly select channels and generate traffic for the Wi-Fi APs constrained such that $S_{avg} < 0.5$. We show the resulting algorithm performance for the low similarity scenario in Figure 6a.

We find that in a low similarity scenario, the greedy selection algorithm and SOFI end up performing particularly well. This is because a low similarity metric is generally only achievable when the Wi-Fi channel occupancy tends to be vastly different between neighboring eNBs across channels and the channel with the lowest Wi-Fi occupancy in each eNB will tend to be different from its neighbors. In such a situation, the best channel for each eNB ends up being the one with the lowest measured Wi-Fi occupancy locally, which is ideal for the greedy algorithm and easily discovered by SOFI as well. However, SOPI ends up under-performing due to the probabilistic selection, since SOPI lacks sufficient information to guarantee neighboring eNBs will select the channel with the most local availability. Romero-Q's performance is below greedy simply because of the learning ramp-up.

b) High similarity: For the high similarity scenario, we modify the fixed topology from the low similarity analysis such that all Wi-Fi APs are deployed in a small radius around the environment center. This placement results in a high dependence between channel access probabilities, as each Wi-Fi

(a) Algorithm performance in a low similarity scenario.

(b) High similarity with low average Wi-Fi utilization.

(c) High similarity with high average Wi-Fi utilization: saturated channel conditions where improvements are critical.

(d) High similarity with independent channel access probabilities.

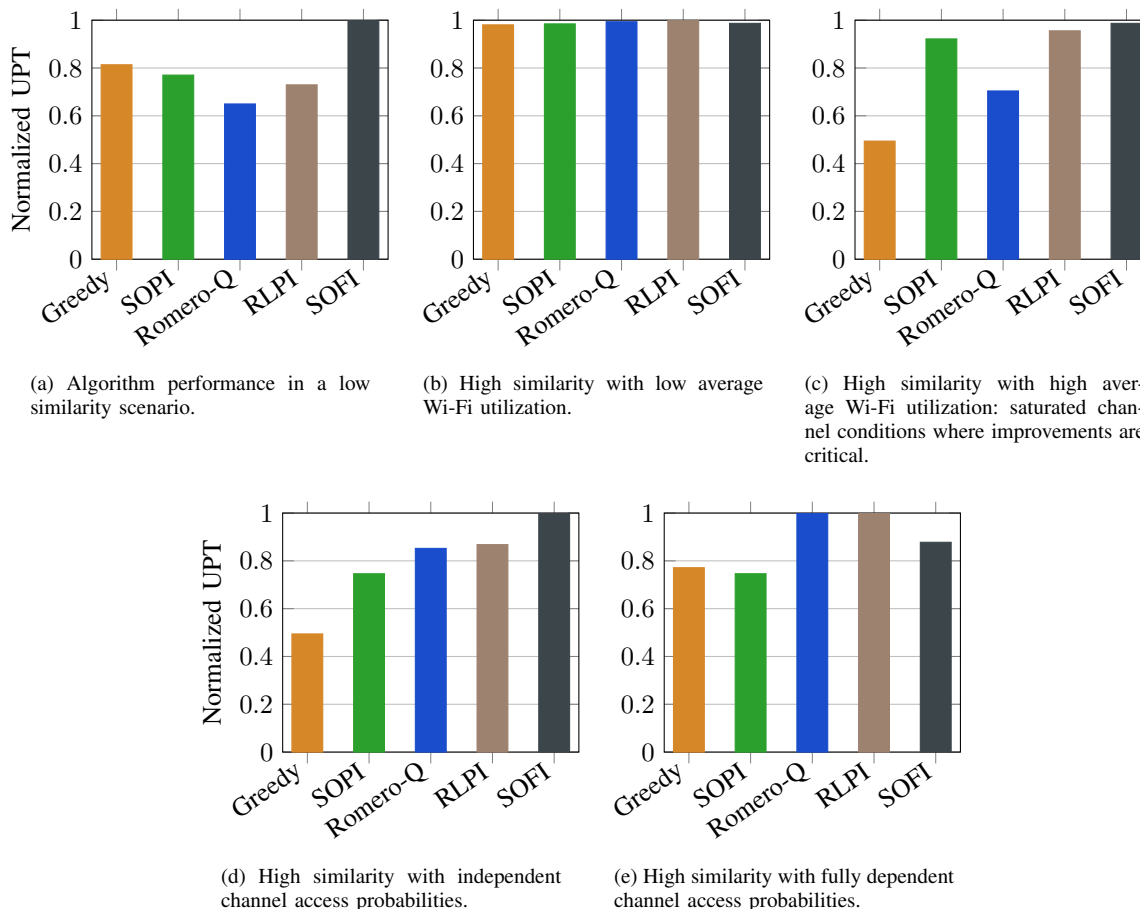(e) High similarity with fully dependent channel access probabilities.

Figure 6. Simulations evaluating the algorithm performance in various scenarios with high/low similarity and Wi-Fi utilization.

AP placed has a high probability of impacting multiple eNBs. We reduce the total number of Wi-Fi APs to better control average channel utilization while still randomizing traffic and Wi-Fi AP channel assignments. An example deployment under this fixed topology constraint is shown in Figure 4c.

Additionally, we simulate the high similarity scenario with two different sub-conditions: low and high average Wi-Fi utilization. We label the simulation realization as having a low average Wi-Fi utilization when the average utilization experienced at each eNB over all channels is under 10%. Conversely, we label the simulation realization as having a high Wi-Fi utilization when the average utilization is over 50%. We show the resulting algorithm performance for both low and high Wi-Fi utilization settings in Figures 6b and 6c.

When the Wi-Fi utilization is not a factor due to low occupancy, none of the eNBs have trouble offloading all their traffic. This is true even if there is some frequency overlap between them. This situation can be considered an abundance of spectral resources, and thus the UPT is similarly high across all channel selection algorithms. However, as licensed cellular providers and a myriad of new Internet of Things (IoT) devices pop up in unlicensed spectrum, the more likely (and important) situation is one in which the spectrum is saturated.

When spectral resources are scarce in the high Wi-Fi utiliza-

tion case, proper channel selection becomes vital to achieving serviceable performance. In this situation, slight differences in channel availability between channels can have a significant impact on expected capacity. The SOPI, RLPI, and SOFI algorithms can evaluate and exploit these differences to great effect. Our results show that these algorithms significantly outperform Romero-Q and Greedy, demonstrating one of the major benefits of sharing the Wi-Fi utilization information between eNBs.

When considering the impact of Wi-Fi utilization similarities between nodes, we can further differentiate between those that are similar due to interference from the same Wi-Fi APs and those that have different Wi-Fi APs, but similar traffic statistics. The impact of these differences is most noticeable in the channel access probability and the resulting channel capacity calculation of the SOPI algorithm in Equations 4 and 5. The presented equation assumes independent channel access probabilities between each eNB and its neighbors, which is achievable when different Wi-Fi APs within the sensing range of each eNB. However, when the Wi-Fi APs are shared, the neighboring channel access probabilities should instead be conditioned on i. This error leads to a differential in capture probability and subsequent capacity calculation depending on the dependence between neighboring eNB access probabilities.

The impact of this error can be seen in Figures 6d and 6e, where we compare the performance of each channel access algorithm given fully dependent or fully independent channel access probabilities between eNBs. We generate the independent access probability scenario using the topology shown in Figure 4b and dependent access probability scenario using the topology in Figure 4c. Both scenarios are crafted with random Wi-Fi traffic generation.

In the fully dependent scenario, the SOPI and SOFI algorithms overestimate their channel capture probabilities due to the assumption of independence and end up being overshadowed by the reinforcement based algorithms. Conversely, the SOFI algorithm ends up outperforming all other algorithms when the access probabilities are independent, and the joint optimization between eNBs can accurately calculate expected capacities.

Overall, we see that while the context-reliant algorithms are all impacted by the access probability dependencies, the reinforcement learning algorithms remain unaffected. Alternatively, context-reliant algorithms perform better when spectral resources are scarce. Our proposed RLPI algorithm balances these two approaches and achieves high performance across each of the scenarios studied, and results in improved performance in the general case as well.

V. Conclusions

In this paper, we examined uLTE coexistence with other uLTE eNBs in the presence of spatially heterogeneous Wi-Fi utilization. While other similar works consider uniform Wi-Fi behavior or ignore Wi-Fi presence entirely, our work used knowledge of Wi-Fi incumbency to outperform existing solutions. To do so, we first formulated a framework under which neighboring eNBs can share Wi-Fi utilization estimates with each other. Next, we presented a unique algorithm in which deployed eNBs exploit the shared utilization estimates to select an optimal transmission channel. We then combined this analytical model with state-of-the-art reinforcement learning techniques and presented a novel reinforcement learning approach (RLPI) that leverages the shared utilization statistics. Finally, we simulated five relevant channel selection algorithms, showing significant improvements in performance in both general random deployments and scenarios with high spectral congestion.

References

[1] Qualcomm Incorporated, "Extending LTE Advanced to unlicensed spectrum," Tech. Rep., Dec 2013.

[2] T. Novlan, B. L. Ng, H. Si, and J. C. Zhang, "Overview and Evaluation of Licensed Assisted Access for LTE-Advanced," in *2015 49th Asilomar Conference on Signals, Systems and Computers*, Nov 2015, pp. 1031–1035.

[3] 3GPP, "Study on Licensed-Assisted Access to Unlicensed Spectrum;," 3rd Generation Partnership Project (3GPP), Technical Report (TR) 36.889, 06 2015, version 13.0.0.

[4] T-Mobile, "T-Mobile Completes Nation's First Live Commercial Network Test of License Assisted Access (LAA)," Jun 2017. [Online]. Available: https://www.t-mobile.com/news/lte-u

[5] K. Hill, "GSA Sees Momentum for CBRS Market and use of LTE in Unlicensed Spectrum," May 2019. [Online]. Available: https://www.rcrwireless.com/20190509/network-infrastructure/gsa-sees-momentum-for-cbrs-market-and-use-of-lte-in-unlicensed-spectrum

[6] M. Haider and M. Erol-Kantarci, "Enhanced LBT Mechanism for LTE-Unlicensed Using Reinforcement Learning," in *2018 IEEE Canadian Conference on Electrical Computer Engineering (CCECE)*, May 2018, pp. 1–4.

[7] Qualcomm Technologies Inc., "Qualcomm Research LTE in Unlicensed Spectrum: Harmonious Coexistence with Wi-Fi," Jun 2014. [Online]. Available: https://www.qualcomm.com/media/documents/files/lte-unlicensed-coexistence-whitepaper.pdf

[8] H. Li, Y. Chang, F. Hao, A. Men, J. Zhang, and W. Quan, "Study on dynamic channel switch in license-assisted-access based on listen-before-talk," in *2016 International Symposium on Wireless Communication Systems (ISWCS)*. IEEE, 2016, pp. 506–510.

[9] S. F. Thomas Cheng and D. Larsson, "Licensed Assisted Access: Practical Coexistence Solutions," Feb 2015. [Online]. Available: https://www.ericsson.com/en/blog/2015/2/licensed-assisted-access-practical-coexistence-solutions

[10] O. Sallent, J. Pérez-Romero, R. Ferrús, and R. Agustí, "Learning-based coexistence for lte operation in unlicensed bands," in *2015 IEEE International Conference on Communication Workshop (ICCW)*, June 2015, pp. 2307–2313.

[11] R. S. Sutton, A. G. Barto *et al.*, *Introduction to reinforcement learning*. MIT press Cambridge, 1998, vol. 135.

[12] S.-H. Cha, "Comprehensive survey on distance/similarity measures between probability density functions," *City*, vol. 1, no. 2, p. 1, 2007.

[13] *LTE-U CSAT Procedure TS V1.0*, LTE-U Forum, 10 2015.

[14] E. Almeida, A. M. Cavalcante, R. C. D. Paiva, F. S. Chaves, F. M. Abinader, R. D. Vieira, S. Choudhury, E. Tuomaala, and K. Doppler, "Enabling LTE/WiFi coexistence by LTE blank subframe allocation," in *2013 IEEE International Conference on Communications (ICC)*, June 2013, pp. 5083–5088.

[15] A. Castañé, J. Pérez-Romero, and O. Sallent, "On the Implementation of Channel Selection for LTE in Unlicensed Bands using Q-learning and Game Theory Algorithms," in *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)*, June 2017, pp. 1096–1101.

[16] J. Perez-Romero, O. Sallent, R. Ferrus, and R. Agusti, "A robustness analysis of learning-based coexistence mechanisms for lte-u operation in non-stationary conditions," in *2015 IEEE 82nd Vehicular Technology Conference (VTC2015-Fall)*, Sep. 2015, pp. 1–5.

[17] H. Ko, J. Lee, and S. Pack, "Joint optimization of channel selection and frame scheduling for coexistence of lte and wlan," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 7, pp. 6481–6491, July 2018.

[18] U. Challita, L. Dong, and W. Saad, "Proactive resource management for lte in unlicensed spectrum: A deep learning perspective," *IEEE Transactions on Wireless Communications*, vol. 17, no. 7, pp. 4674–4689, July 2018.

[19] F. Liu, E. Bala, E. Erkip, M. C. Beluri, and R. Yang, "Small-cell traffic balancing over licensed and unlicensed bands," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 12, pp. 5850–5865, Dec 2015.