基于深度强化学习的多单元网络功率分配

张勇，康灿平，马腾腾，滕颖蕾，郭达

北京邮电大学电子工程学院，北京，Email: {*Yongzhang, canping kang*, mtt, lilytengtt, guoda}@bupt.edu.cn

摘要-本文对多小区功率分配方法进行了研究。与传统的优化分解方法不同，采用深度强化学习(Deep Reinforcement Learning, DRL)方法来解决NP-hard问题中的功率分配问题。我们的工作目标是在基站随机且密集分布的场景下，最大化整个网络的整体容量。我们提出了一种无线资源映射方法和一种用于多单元功率分配的深度神经网络，称为deep - q - full - connected - network (DQFCNet)。与充水功率分配和Q-learning方法相比，DQFCNet可以实现更高的总容量。此外，仿真结果表明，DQFCNet在收敛速度和稳定性方面有显著提高。

关键词:深度学习，强化学习，功率分配，资源分配，多单元网络，功率控制

I. 介绍

近十年来，人们一直在研究LTE系统和5G移动通信系统中的功率和无线资源分配问题。随着无线通信的快速发展，网络的规模越来越大。特别是在超密集网络(Ultra Dense Network, UDN)中，基站(Base station, BSs)数量急剧增加。功率和无线资源的最优分配问题变得尤为尖锐。不恰当的分配方案降低了网络频谱效率。

多单元资源分配是一个NP-hard问题[1]。目前，很多研究者都在研究单小区环境下的资源分配问题，包括异构网络[2]-[4]。然而，小区间干扰增加了问题的复杂性。提高一个BS的发射功率可以提高其容量，但会降低相邻小区的容量。任何BS都不能单方面修改自己的发射功率来提高全网的总速率。实现自组织的主要方法可以分为两大类:学习和优化[5]。

在过去的几年里，优化成为了最流行的方法，在很多文献中都出现了[6]-[9]。参考文献[6]-[8]提出了联合资源分配和功率控制方案，以获得最大的能效。作为通信网络最基本的问题，我们更关心的是整个网络的整体容量。在不优化功率分配机制的情况下，贾石等人设计了子载波分配算法，以减轻六边形LTE/LTE- a网络中的小区间干扰(ICI)并提高频谱效率[9]。参考[10]在同一网络中提出了一种协作的ICI协调技术，相邻的enodeb相互协商，根据用户满意度函数确定发射功率。解决方案很简单，但作者没有验证解决方案在大规模网络中的收敛性。参考文献[1]研究了在六边形网络中最大化加权和速率的协调调度和功率控制方法。参考文献[11]将该问题分解为基于图划分的子问题，并采用最优和启发式方法进行求解。

利用学习方法解决多小区网络中的联合功率和资源分配问题是一种新颖的方法。关于多小区网络中的资源分配问题已经做了一些工作。Wang等人研究了室内企业封闭接入小型BS网络的发射功率问题[13]。Usama等人提出了一种多参数Q-Learning算法，利用子带功率因数和边缘到中心边界来最小化ICI，最大化总信噪比(SINR)[14]。但是，整个网络的SINR最大化并不意味着整个网络的总容量最大化。

受深度强化学习(deep reinforcement learning, DRL)成功的启发[15]，我们提出了一种基于DRL的多单元网络的功率分配方法。我们根据无线网络的特点，将深度强化学习引入无线网络中。首先，考虑到无线网络的巨大状态空间，对发射功率进行离散化，并从系统的收敛性和稳定性方面分析离散粒度的影响。其次，提出动态状态添加策略，降低计算复杂度。最后，介绍了深度神经网络和目标函数，以及速度

978-1-5386-6358-5/18/$31.00 ©2018 IEEE

加快收敛速度。通过数值仿真验证了我们的算法能够提高网络的频谱效率。

II. 系统模型与问题表述

A. 系统模型

该系统被认为是具有密集部署的BSs的下行多小区OFDM小区。系统采用reuse-1模型[16]，即每个BS都可以使用所有可用带宽。网络模型与[11]相同，如图1所示。一个中央控制器可以收集整个网络的信息，包括SINR和发射功率。∑BS n在时刻t的{用户数}为 $usnt \in Mt1, \cdots Mtn, \cdots, MtN$ ($Nn=1\ Mtn = M$),其中为用户总数。我们的工作采用随机游走模型。用户在t时刻的移动速度为$V_{tm}(0 \leqslant Vtm \leqslant V\ max)$，移动角度为$D_t^m(0 \leqslant Dtm \leqslant 2\pi)$。BSs的地理分布遵循泊松点过程模型。每个BS$BS_n^f$完全重用K正交子载波。当用户连接到BS时，BS被激活，发射功率为pnt。每个用户在时刻t只能连接到一个BS。每个子载波只能分配给一个用户。使用六径衰落信道模型进行评估。信道模型的选择并不影响我们建议的性能。

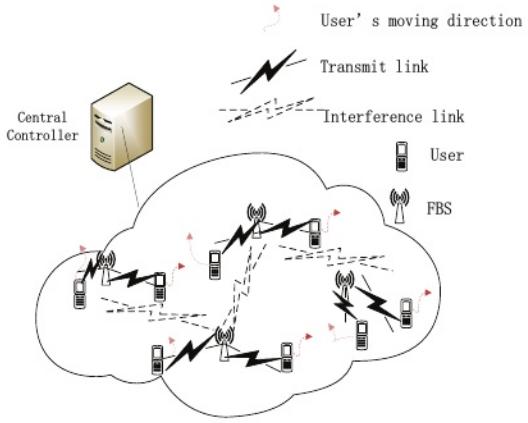


图1所示。系统模型。

B. 问题公式化

令$\eta(n,k,m)t$表示第n个BS服务用户m在时刻t在第k个子载波上的接收SINR，其中$k \in \{1, \cdots, K\}$，由给出

$$\eta_t^{(n,k,m)} = a_t^{(n,m)} \alpha_t^{(n,k,m)} \frac{g_{n,m}^{(k)} p_t^{(n,k)}}{\sum_{n' \neq n} g_{n',m}^{(k)} p_t^{(n',k)} + \sigma^2} \quad (1)$$

式中$gn,m_{(k)}$和$gn(k)',m$表示用户m在第k个子载波上的第n个和n'-个BSs的信道增益，分别为-(n,k) $_{(n',\ k)}$有效。p_t和p_t分别表示第k个子载波上n个BSs的总发射功率。$\alpha t(n,k,m)$表示BS n是否将子载波k分配给用户m $\alpha t(n,k,m) \in [0,1]$，$\sigma 2$表示高斯白噪声的幂。$At\ (n,m)$表示是否为用户m

连接到BS n:

$$a_t^{(n,m)} = \begin{cases} 1, & \mathrm{m} \in \mathrm{M_i} \\ 0, & \mathrm{m} \notin \mathrm{M_i} \end{cases} \quad (2)$$

系统性能根据测量的总容量(bps/Hz)进行分析。BS_n^f在时间t时其关联用户在子载波k上所实现的容量由:

$$C_t^{(n,k)} = \frac{B}{K} \log_2 (1 + \sum_{m=1}^{M} \eta_t^{(n,k,m)}) \quad (3)$$

总体容量可以定义为:

$$R_t = \sum_{n=1}^{N} \sum_{k=1}^{K} C_t^{(n,k)} \quad (4)$$

我们的目标是基于近乎最优的子载波分配，通过调整子载波$pnt = [\mathrm{pt(n,1)} \cdots p(n,k)_{\ t \cdots p(n,k)\ \mathrm{t}}]$上BSs的发射功率来增加整个网络的总容量。优化问题可以表述为:

$$\begin{cases} \boxtimes \\ \arg\max R_t \\ s.t.C1: p_t^{(n,k)} \geq p_{\min}, \forall n,k \\ \quad C2: \sum_{n,k} p_t^{(n,k)} \leq p_{\max}, \forall n,k \end{cases} \quad (5)$$

式中p_{max}为BS的最大发射功率，p_{min}为副载波的最小发射功率。

在网络初始化阶段，用户按照最大SINR准则完成与基站的关联。用户与基站之间的干扰主要来自小区间干扰，小区内用户之间不存在干扰。链路速率受附加基站到用户的信号强度、小区间干扰强度的影响。我们的目标是调整基站的发射功率，以增加整个网络的整体容量。考虑到用户的公平性，我们将一个基站的下行子载波平均分配给所有附属用户。子载波上的功率根据注水算法进行初步分配。

上述问题是一个多目标非凸优化问题。解决这个问题的传统方法是启发式搜索算法。然而，这些算法大多是非常低效的

运行时间长，无法在线实时调整。为了解决这个问题，在下一章中，我们将介绍一种深度强化学习算法来解决这个问题。

III. DEEP-Q-FULL-CONNECTED-NETWORK解决方案

A. 强化学习-多智能体Q学习

分布式认知BSs的场景可以用随机博弈[17]来数学表述，其中每个决策系统的学习过程可以用五元组{N, S, A, P, R(S, A)}来表示，其中:

- $_{n}$ ={1, 2，⋯，N_f}:为代理的集合(即:;BSs)。
- $_{S}$ = {s_1，s_2，⋯，S_m}:是系统可以占据的可能状态的集合，其中m是可能状态的数量。
- $_{A}$ = {A_1，A_2，⋯，a_l}:是一组可能的动作
- $_{P}$:是概率转移函数，表示系统从一种状态转移到另一种状态的概率。
- $_{R(s,⊠a)}$:定义奖励函数，当在状态s∈s中执行联合动作⊠a时，确定对agent n的奖励。

q学习作为一种基于值函数的无模型强化学习算法，在解决动态无线网络环境问题方面具有天然的优势。在q学习算法中，每个agent通过不断的迭代学习收敛自己的行为值函数。这个行为值函数一般用一个表表示$Q(s_m, a_l)$，其中a_l∈a，s_m∈S。因此，这个表的大小为$m \times 1$。q值$Q(s_m, a_l)$表示动作a_l在状态s_m的累积奖励在无限时间范围内的期望值，由下式给出:

$$Q_\pi(s,a) = E_\pi[\sum_{k=0}^{\infty} R_{t+1} + \gamma Q(S_{t+1}, A_{t+1}) | S_t = s, A_t = a] \tag{6}$$

其中$R_{(t+1)}$表示在状态s下采取行动a后的即时奖励，0⩽γ⩽1为折现系数，表示经验值的权重。

q值更新如下:

$$Q_n^{t+1}(s_m^n, a_l^n) := (1-\alpha) Q_n^t(s_m^n, a_l^n) + \alpha(R_n^t + \gamma \max_{a' \in A} Q_n^t(s_{m'}^n, a')) \tag{7}$$

其中0⩽α⩽1表示学习率。

B. 功率分配算法和Q学习映射

提出的基于功率分配的q学习算法是一种基于与环境持续交互的多智能体算法。智能体、状态、动作和奖励函数定义如下:

- $_{代理}$:BS n, 1⩽n⩽N_f。
- $_{状态}$:$s_t^{n,k}$={M_t^n，pnt}，其中M_t^n表示时刻t连接到BS n的用户数量，pnt表示时刻n个BS的幂。为了降低算法的复杂度和网络的状态空间，对BS的幂进行了离散化处理。离散规则如下:

$$p_t^n = \tau, (P_{\max}^f - A_\tau) \le \sum_{k=0}^{K} p_t^{n,k} < (P_{\max}^f - A_{\tau+1}) \tag{8}$$

其中τ∈{0,1,2,3,4,5}，A_0=p^f_{max}，A_6= 0，和A1 A5是任意选择的阈值。

- $_{作用}$:an = {kn, p∈$t(n,k)$}，其中kn表示第n个BS的第k个子载波。p⊠t (n, k) de -

注意到第n个BS的第k个子载波上功率的调整值，p∈$t(n,k)$∈{−|∈$p(n,k)t$|， 0， +|∈$p(n,k)t$|}是由采取行动后h个用户在当前用户周围的总吞吐量C_t^h的变化决定的。如果C_t^h增大，则p⊠$t(n,k)$ = +|⊠$pt(n, k)$|，反之亦然。

- $_{奖励}$:第k个子载波上第n个BS的奖励定义为[18]:

$$r_t^{n,k} = \begin{cases} 1 - e^{-(C_t^{(n,k)})}, & \sum_{k=1}^{K} p_t^{(n,k)} \le P_{\max}^f \\ -1, & Otherwise \end{cases} \tag{9}$$

由于环境的动态性，环境的状态空间比较大，每个agent的q值表的大小也不一样。如果我们确定一个固定大小的大q值表，计算复杂度将会高得多。因此，引入了动态状态添加方法，即当有新的状态时，这个状态会自动添加到状态集合中。无需为每个智能体定制Q表，这是动态方法的优点。并且可以提高Q表的查找效率，提高存储空间利用率。但这也意味着Q表需要很长的时间来收敛，因为当新的状态发生时，网络需要重新收敛。因此，本文提出了一种DRL算法来加快q值表的收敛速度。

C. DQFCNet

DeepMind提出了[15]的概念。与前面介绍的Q-learning相比，值函数的更新是神经网络的参数，而不是q值表。更新方法采用梯度下降算法。值函数更新如下:

$$\theta_{t+1} = \theta_t + \alpha[r + \gamma \max_{a'} Q(s', a'; \theta^-) - Q(s, a; \theta)] \nabla Q(s, a; \theta) \tag{10}$$

其中r+γ max $Q(s'$， a';θ−)为时间差$_{a'}$(TD)的目标。$问(牟代,;\theta)$是通过值函数逼近的网络对象。和$\nabla Q(s, a;\theta)$为梯度。

在很多游戏中，比如围棋，DQN都取得了不错的效果。游戏环境以无噪音为基础，

完全信息状态。然而，无线网络环境存在噪声、不完全信息和软边界。很难将这些信息映射到DQN中。因此，设计了简化版的DQN (DQFCNet)，通过离散环境状态数和使用深度全连接神经网络来加速行为值函数的收敛。

首先，提出了如图2所示的深度全连接神经网络，该网络包含四层，输入层数据为

$$inputs_t = [M_t^1, \dots M_t^n, ..., M_t^N, {p_t}^{(1,1)}, \dots, {p_t}^{(n,k)}, \dots, {p_t}^{(N,K)}, {csi_t}^{(a_{n,1},k)}, \dots {csi_t}^{a_{n,m},k}, ..., {csi_t}^{(a_{n,M},K)}] \quad (11)$$

其中$csi_{ta^{n,m},k}$表示第n个BS的第k个子载波上的第m个用户的信道状态信息。中间两层隐藏层主要是增加网络优化的非线性度，提高网络的拟合能力。输出层数据为:

$$outputs = [|\vec{p}_t^{(1,1)}|, ..., |\vec{p}_t^{(n,k)}|, ..., |\vec{p}_t^{(N,K)}|] \quad (12)$$

式中$|\in pt(n,k)|$表示第k个子载波上的第n个BS的总功率调整值。

深度神经网络采用dropout技术[19]，增加网络泛化能力，减少网络方差，防止过拟合。为了加快网络的训练速度，在网络的反向传播中使用了AdamOptimizers[20]。深度神经网络的损失函数设计如下:

$$Loss = \frac{1}{s}\sum_{i=0}^{s}(q_z - o_z)^2 + 1/mean(o_z) + c||\theta||_2 \quad (13)$$

式(11)，q_z表示Q学习调整策略。o_z表示神经网络的输出，即每个BS在每个子载波上的总功率调整值。C为惩罚因子。θ表示权重参数。损失函数的第一部分是使神经网络的输出尽可能接近q值表。第二部分是输出值均值的倒数。当损失函数最小时，网络会尝试提高BSs的平均功率调整，这样可以有效地提高整个网络的整体容量。第三部分是l2范数，它的作用是对参数θ做一些限制，从而防止网络过拟合。

在训练过程中，对神经网络模型f进行评估。如果f的网络调整能力强于当前最佳模型f^*，则将当前模型f作为最佳模型f^*来增强Q学习。

为了加快收敛过程，提高整个网络的频谱效率，输出的

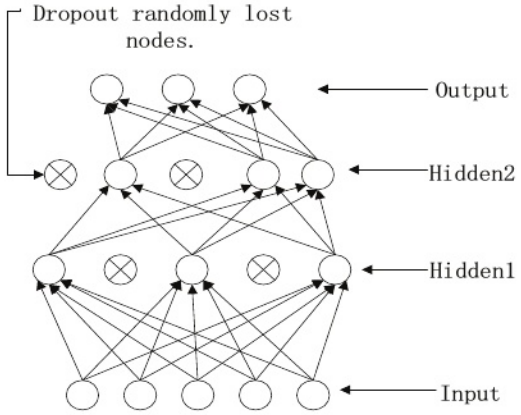


图2所示。深度神经网络模型。

引入深度神经网络来加速动作选择(等式13)。

$$a_n^k = \arg\max_{a\in A}((1-\beta)*\sum_{1<n'<N_f} Q_{n'}^k(s^{n',k},:)+\beta*f_n^*) \quad (14)$$

其中fn *表示当前最佳深度神经网络模型在第n个BS上的调整策略，起策略增强作用。$其中\beta$用于调整神经网络输出的侧重点。在网络训练的初始阶段，要多注意神经网络的输出，这样β应该比较大。当智能体继续学习时，q值表的权重应该变得更大。

D. 复杂性分析

将DQFCNet的计算复杂度与文献[11]中的算法进行了比较。在[11]中，分支切断算法用于寻找多小区场景下整个网络总容量的上界。主要的计算开销由分支切断算法产生。假设环境状态用长度为m的向量表示，这种方法的计算复杂度为$O2^m$。在本文提出的深度神经网络中，激活函数使用relu，模型训练完成后，算法的复杂度为:$O(m*(n1*n2 + n2*n3 + n3*n4))$，其中$n1$、$n2$、$n3$、$n4$是每层神经元的数量。从上面的分析可以看出，DQFCNet大大降低了算法的复杂度。

IV. 仿真与评估

整个蜂窝网络环境包括N_fBSs。每个用户连接到信噪比最高的BS。所有BSs共享频谱带宽。在仿真中，我们设置噪声功率σ^2=10^{-7}，每个BS在每个子载波上的功率调节幅度，其中A1-A5=[25,20,15,10,5]。其他一些参数设置如表I所示，β更新为表II。和

表一仿真参数

Parameters	Values
Bandwidth B	10 MHz
The number of BSs	32
The number of UEs	200
Radius r	15 m
Initial power p	0-5 dbm
Network Size	100*100 m^2
$p_{\max}$	30 dbm
$p_{\min}$	-100 dbm
$v_{\max}$	1m/s
h	5
α	0.5
γ	0.9
ε	0.2
Dropout rate	0.8

表2 β更新

Parameters	Values
Iteration step (0-30)	0.8
Iteration step (30-60)	0.6
Iteration step (60-100)	0.3
Iteration step (100-)	0.1

在每一步，由于节点的移动，网络更新一次拓扑结构。

在深度神经网络中，每层神经元的数量被配置为[14880,10000,5000,2048]。训练参数数量为209,057,048。

图3显示了模型收敛后三种模式的频率效率。每一次迭代，用户都会移动，网络拓扑结构也会更新。由图3可知，DQFCNet的频率效率最高，且比Q-learning和充水算法更稳定。我们可以看到，DQFCNet算法正在接近[11]中提到的算法的上界。

网络的收敛情况如图4所示。由于节点的迁移性，Q-learning的波动很大。当网络拓扑发生变化时，Q-learning需要重新计算和收敛。在动态场景中，虽然DQFCNet也会有波动，但相对于Q-learning来说，它是比较稳定的。同时，随着深度神经网络策略的加强，DQFCNet大大提高了频谱效率。

图5为功率离散化为三个层次的收敛对比图。从图中可以看出，在不同离散粒度下，整个网络的收敛速度非常小。但可以明显发现，当离散粒度为6时，整个网络更加稳定，频谱效率更高。如果将功率离散成15，则环境态的数量过大。模型收敛后，系统稳定，波动小。当色散为3时，模型收敛后出现波动

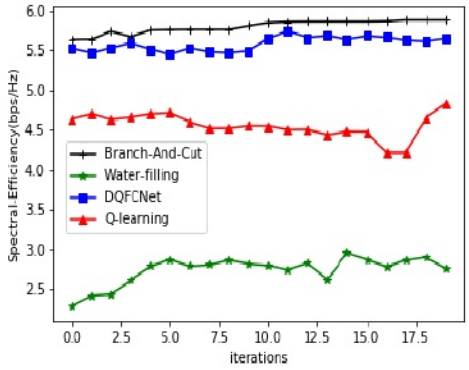
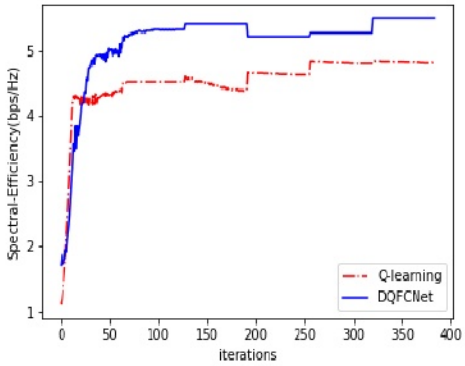


图3所示。频谱效率比较(模型已经收敛)。

图4所示。收敛速度比较。



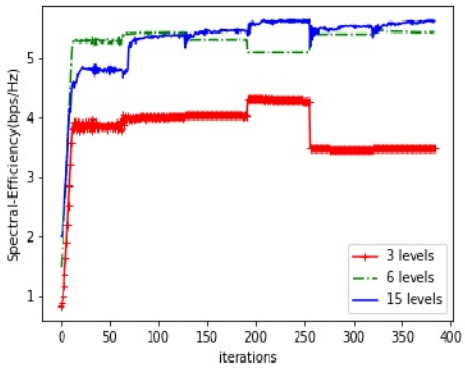


图5所示。不同离散粒度上收敛性比较。

很大程度上是因为离散粒度太粗，不能代表环境的真实变化。

V. 结论

多小区功率分配问题是一个非凸问题，一直是研究的难点

无线通信领域。传统的求解方法是最优化分解，但计算复杂度高。受深度学习领域研究成果的启发，我们提出了DQFCNet方法，将深度强化学习应用于多单元功率分配，取得了良好的系统容量增益。此外，由于使用了深度神经网络，DQFCNet的收敛速度和稳定性得到了提高。

深度学习的成功对于多小区无线网络优化的研究具有指导意义。需要指出的是，虽然本文的目标是最大化整个网络的整体容量，但我们的建议也可以推广到其他技术指标，如系统能效、体验质量等。

鸣谢

本研究得到国家自然科学基金资助(批准号: 61771072)。

参考文献

[1] Zhang, Honghai, et al. ”Weighted Sum-Rate Maximization in Multi-Cell Networks via Coordinated Scheduling and Discrete Power Control.” IEEE Journal on Selected Areas in Communi-cations 29.6 (2011):1214-1224.

[2] Yang, Kai, et al. ”Energy-Efficient Downlink Resource Allocation in Heterogeneous OFDMA Networks.” IEEE Transactions on Vehicular Technology 66.6(2017):5086-5098.

[3] Yang, Zhaohui, et al. ”User Association, Resource Allocation and Power Control in Load-Coupled Heterogeneous Networks. ” GLOBECOM Workshops IEEE, 2017:1-7.

[4] Zou, Shangzhang, et al. ”Joint Power and Resource Allocation for Non-Uniform Topologies in Heterogeneous Networks. ” Vehicular Technology Conference IEEE, 2016:1-5.

[5] Aliu, Osianoh Glenn, et al. ”A Survey of Self Organisation in Future Cellular Networks.” IEEE Communications Surveys & Tutorials 15.1 (2013):336-361.

[6] Wang, Xiaoming, et al. ”Energy-Efficient Resource Allocation in Coordinated Downlink Multicell OFDMA Systems. ” IEEE Transactions on Vehicular Technology 65.3(2016):1395-1408.

[7] Lahoud, Samer, et al. ”Energy-Efficient Joint Scheduling and Power Control in Multi-Cell Wireless Networks.” IEEE Journal on Selected Areas in Communications 34.12(2016):3409-3426.

[8] Yang, Kai, et al. ”Energy-Efficient Downlink Resource Allocation in Heterogeneous OFDMA Networks.” IEEE Transactions on Vehicular Technology 66.6(2017):5086-5098.

[9] Shi, Jia, L. L. Yang, and Q. Ni. ”Novel Intercell Interference Mitigation Algorithms for Multicell OFDMA Systems With Lim-ited Base Station Cooperation. ” IEEE Transactions on Vehicular Technology 66. 1 (2017):406-420.

[10] Yassin, Mohamad, et al. ”Cooperative resource management and power allocation for multiuser OFDMA networks. ” Iet Communi-cations 11.16(2017):2552-2559.

[11] Pateromichelakis, Emmanouil, et al. ”Graph-Based Multicell Scheduling in OFDMA-Based Small Cell Networks.” IEEE Ac-cess 2 (2014):897-908.

[12] Zhang, Chaoyun, P. Patras, and H. Haddadi. ”Deep Learning in Mobile and Wireless Networking: A Survey.” (2018).

[13] Wang, Zhiyang, et al. ”Learn to adapt: Self-optimizing small cell transmit power with correlated bandit learning.” IEEE Inter-national Conference on Communications IEEE, 2017:1-6.

[14] Sallakh, Usama, S. S. Mwanje, and A. Mitschele-Thiel. ”Multi-parameter Q-Learning for downlink Inter-Cell Interference Co-ordination in LTE SON.” Computers and Communication IEEE, 2014: 1-6.

[15] Mnih, V, et al. ”Human-level control through deep reinforcement learning. ”Nature 518.7540(2015):529.

[16] Mazarico, Juan I, et al. ”Detection of synchronization signals in reuse-1 LTE networks.” Wireless Days IEEE, 2009:1-5.

[17] Burkov, Andriy, and B. Chaib-Draa. ”Labeled initialized adaptive play Q-learning for stochastic games.” Draa (2010).

[18] Saad, H, A. Mohamed, and T. Elbatt. ”A cooperative Q-learning approach for distributed resource allocation in multi-user femtocell networks. ” Wireless Communications and NETWORKING Con-ference IEEE, 2014:1490-1495.

[19] Srivastava, Nitish, et al. ”Dropout: a simple way to prevent neural networks from overfitting.” Journal of Machine Learning Research 15. 1(2014):1929-1958.

[20] Kingma, Diederik P, and J. Ba. ”Adam: A Method for Stochastic Optimization.” Computer Science (2014).