A Study about Heterogeneous Network Issues Management based on Enhanced Inter-cell Interference Coordination and Machine Learning Algorithms

By:

Weijie Qi

A thesis submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy

The University of Sheffield
Faculty of Engineering
Department of Electronic and Electrical Engineering

Submission Date

14/03/2018

Acknowledgment

During the whole period of research work, my parents Guorui Qi and Changxiu Zhou have provided all the help I needed from them – LOVE. I also owe my thanks to my wife, Jia Sun, who quietly supports me and encourages me all the time.

I want to acknowledge the help from my supervisor, Prof. Jie Zhang. With his guide and experience in not only academic but also daily life, I have managed to overcome the difficulty and made research achievement. I also want to thank Dr. Xiaoli Chu for her instruction on my research field. My thanks are also given to my colleagues and friends, who support me with my research work, Baoling Zhang, Youchen Wang, Haonan Hu, Hao Li, Qi Hong.

Finally, I want to thank Dr. Thomas Walther and Hilary J Levesley for their help and assistance for my thesis writing and all the staffs that help me finish my paper work during my PhD period.

Weijie Qi

2018

Abstract

Under the circumstance of fast growing demands for mobile data, Heterogeneous Networks (HetNets) has been considered as one of the key technologies to solve 1000 times mobile data challenge in the coming decade. Although the unique multi-tier topology of HetNets has achieved high spectrum efficiency and enhanced Quality of Service (QoS), it also brings a series of critical issues. In this thesis, we present an investigation on understanding the cause of HetNets challenges and provide a research on state of arts techniques to solve three major issues: interference, offloading and handover.

The first issue addressed in the thesis is the cross-tier interference of HetNets. We introduce Almost Blank Subframes (ABS) to free small cell UEs from cross-tier interference, which is the key technique of enhanced Inter-Cell Interference Coordination (eICIC). Nash Bargain Solution (NBS) is applied to optimize ABS ratio and UE partition. Furthermore, we propose a power based multi-layer NBS Algorithm to obtain optimal parameters of Further enhanced Inter-cell Interference Coordination (FeICIC), which significantly improve macrocell efficiency compared to eICIC. This algorithm not only introduces dynamic power ratio but also defined opportunity cost for each layer instead of conventional zero-cost partial fairness. Simulation results show the performance of proposed algorithm may achieve up to 31.4% user throughput gain compared to eICIC and fixed power ratio FeICIC.

This thesis' second focusing issue is offloading problem of HetNets. This includes (1) UE offloading from macro cell and (2) small cell backhaul offloading. For first aspect, we have discussed the capability of machine learning algorithms tackling this challenge and propose the User-Based K-means Algorithm (UBKCA). The proposed algorithm establishes a closed

loop Self-Organization system on our HetNets scenario to maintain desired offloading factor of 50%, with cell edge user factor 17.5% and CRE bias of 8dB. For second part, we further apply machine learning clustering method to establish cache system, which may achieve up to 70.27% hit-ratio and reduce request latency by 60.21% for Youtube scenario. K-Nearest Neighbouring (KNN) is then applied to predict new users' content preference and prove our cache system's suitability. Besides that, we have also proposed a system to predict users' content preference even if the collected data is not complete.

The third part focuses on offloading phase within HetNets. This part detailed discusses CRE's positive effect on mitigating ping-pong handover during UE offloading, and CRE's negative effect on increasing cross-tier interference. And then a modified Markov Chain Process is established to map the handover phases for UE to offload from macro cell to small cell and vice versa. The transition probability of MCP has considered both effects of CRE so that the optimal CRE value for HetNets can be achieved, and result for our scenario is 7dB. The combination of CRE and Handover Margin is also discussed.

List of Publications

Published

Qi, W., Zhang, B., & Zhang, J. (2017). A Markov Based Cell Range Expansion Optimization in Two-tier HetNets System. *International Journal of Engineering Technology, Management and Applied Sciences*, *5*, 17-25.

Qi, W., Zhang, B., Chen, B., & Zhang, J. (2018, January). A user-based K-means clustering offloading algorithm for heterogeneous network. In *Computing and Communication Workshop and Conference (CCWC), 2018 IEEE 8th Annual*(pp. 307-312). IEEE.

Zhang, B., Qi, W., & Zhang, J. (2018, January). An energy efficiency and ping-pong handover ratio optimization in two-tier heterogeneous networks. In *Computing and Communication Workshop and Conference (CCWC), 2018 IEEE 8th Annual* (pp. 532-536). IEEE.

Submitted

Qi, W., & Zhang, A Power-Based Multi-layer Nash Bargain Solution Algorithm for Further Enhanced Inter-cell Interference Coordination Implementation. submitted to *IEEE Communications Letters*

Acronyms

ABS	Almost Blank Subframes
CRE	Cell Range Expansion
CoMP	Coordinated Multi Point
eICIC	enhanced Inter-cell Interference coordination
eNodeBs	Evolved Node B
FAP	Femtocell Access Point
FeICIC	Further enhanced Inter-cell Interference coordination
FFR	Fractional Frequency Reuse
HetNets	Heterogeneous Networks
HBS	Home Base Station
ICIC	Inter-cell Interference coordination
KCA	K-means Clustering Algorithm
KNN	K-Nearest Neighbouring
LTE	Long-Term Evolution
LPN	low power node
MIMO	Multi-Input Multi-Output
MCPs	Markov Chain Processes
ML	Machine Learning
NBS	Nash Bargain Solution
QoS	Quality of Service
RB	Resource Block
RNTP	Relative narrowband transmit power
RRH	Remote Radio Heads
rp-ABS	reducing power ABS
SON	self-organization
TTT	Time-to-Trigger
TTI	Transmit Time Interval
UE	user equipment
3GPP	3rd Generation Partnership Project

Table of Contents

List of Figures

List of Tables

List of Symbols

P	Performance of player in NBS
C	Cost of player in NBS
U	Total Utility in NBS
N	Total Number of UEs in current small cell
N'	Total Number of UEs in current macro cell
N_s	Number of small cells
B	Bandwidth
T	Total number of subframes
r	ABS ratio
a_j	The probability of current UE allocating on normal subframes
b_j	The probability of current UE allocating on ABSs
a_n	Total number of nABS UEs in current small cell
b_n	Total number of ABS UEs in current small cell
c_n	Total number of ABS UEs in current macro cell
P_s	Transmission power of serving small cell
P_m	Transmission power of neighbouring macro cell
P_i	Transmission power of neighbouring interfering small cell
l_s	Path Loss from serving small cell to UE
l_i	Path Loss from neighbouring small cell to UE
l_m	Path Loss from neighbouring macro cell to UE
g_s, g_i, g_m	Fast-fading
d	Distance between serving cell and UE
D	Euclidean distance between two elements within variable space
f	Frequency
a_1, a_2	Ratio of total subframes for two types of rp-ABSs
$b_{1j} \ldots b_{6j}$	The probability of current UE allocating on respect power layer
$n_1 \ldots n_6$	The number of UEs in respect layer
R_m	Received signal power from macro cell
R_s	Received signal power from small cell
R	Training Data set, each element within has *v* parameters
R'	Test Data set, each element within has *v* parameters
N_1	Number of elements within training data

N_2	Number of elements within test data
F1	F_1 score
α	Edge user factor
β	CRE bias
K	Number of Clusters in K-means algorithm
K'	Number of neighbours calculated when applying KNN algorihm
E	System error
$P_M(x)$	Transition probability of UE transit to next M state in MCP
$P_S(x)$	Transition probability of UE transit to next S state in MCP
i, j	Iteration step variable used to indicate the number of repeating steps in loop

Chapter 1. Introduction

1.1 Background and Motivation

The modern consumer's constant demand for mobile broadband is experiencing a rapid increase due to innovations in technologies such as smart phones and mobile devices. Analyst Mason predicted that if operators in Western Europe continue to use traditional macrocell networks to meet the increasing broadband demand, the cost to establish new macrocell BSs will increase to '40 billion USD per year by 2016, compared to 5 billion USD per year in 2011'.

As a result, given the exponential increase in the number and types of user equipment (UE), simply adding Base Stations (BS) to existing networks will not be sufficient to fulfil the modern world's increasing capacity requirements. Therefore, new-generation technology to realise 4G LTE standards is greatly needed. Many advanced technologies, such as carrier aggregation and multi-input multi-output, have been developed and utilised for LTE network implementation in 3GPP Releases 7 and 8, which are still under extensive study [1][2]. Against this backdrop, heterogeneous networks (HetNets) were first developed and applied as an LTE standard in 3GPP Release 10 [3]. Unlike normal networks, which contain only macrocells and remote radio heads (RRHs), HetNets also introduce nodes with lower power (LPN), such as picocell and femtocell networks. These allow HetNets to contain more than two tiers cells. The advantage of HetNets is obvious—these networks enable network mobility and make UE easier to access. The shorter distance between the network and UE reduces signal path loss and increases transmission quality. The larger network scale also increases the spectrum reuse efficiency, thus increasing the rate of data transfer [4]. However, like most new concepts, HetNets also face certain challenges, with interference being one of the most severe ones.

Normally, adjacent cells use different frequencies to guarantee quality of service (QoS) to users. However, in LTE standards, the concept of frequency reuse is introduced and required. Higher frequency reuse means adjacent cells may share more frequencies, thereby increasing capacity. An ideal situation is a frequency reuse ratio of 1, where adjacent cells can share all available frequencies [5]. However, frequency reuse may also result in one resource block (RB) being scheduled to two users in two adjacent cells. The probability of this undesirable scenario increases with the frequency reuse ratio. The consequence is not just a low QoS; instead, like a delay or a lost packet, the unwanted interference caused by the overscheduling of RBs may directly result in radio link failures or call drops.

Thus, though the high frequency reuse strategy of HetNets increases capacity, it also inevitably results in interference; this has become the major issue of HetNets and also cause other special issues related to HetNets [6]. In general, due to HetNets' unique structure, there are mainly three severe challenges need to be solved. Firstly, HetNets inherently involve **cross-tier interference**, which mainly refers to interference signals generated by other-tier networks. In cases of two-tier networks with both macro cells and small cells, small cell UEs may also receive signals from macrocells due to frequency reuse (Figure 1.1). This interference may be severe because macrocells normally have much higher transmission power than small cells [7]. Secondly, HetNets suffer from **offloading** problem due to power differences among different tier of network, which are designed to contain various standard devices. Therefore, UE devices are usually attached to macrocells rather than small cells, due to the macrocells' larger transmission power. This may render HetNets' design or create load unbalance of macro cell and backhaul problem of small cell network [9]. Thirdly, **Ping-pong Handover** problem during offloading phase. When UEs are offloaded from macro cell to small cell, handover will occur. However, the power difference between two cells will generate signal fluctuation at cell edge (normally refers to small cell edge in HetNets), which may lead to frequent Handover happening. Since only control and acknowledge signals are transferred during this process, frequent handover will thus reduce UE's average capacity [10].

In order to realise HetNets' potential, these three major issues should be mitigated. This mitigation requires a new technique of interference management, offloading rebalance and backhaul system.

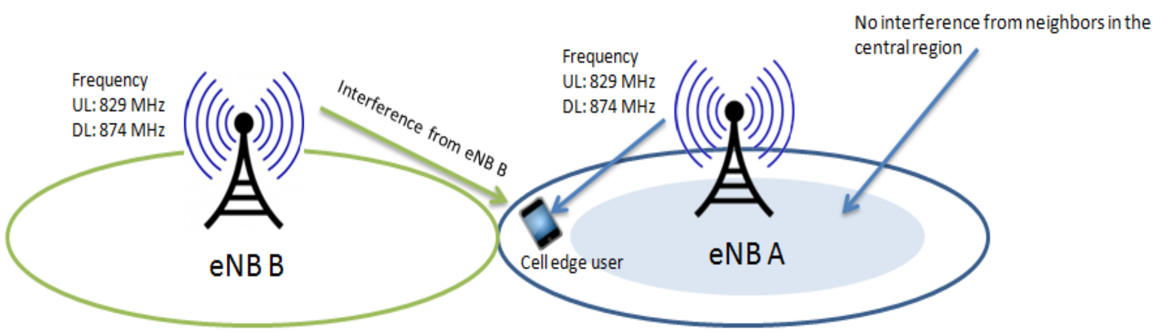

Figure 1-1 Interference Example

For edge users, it may also receive the signal from neighbour cell at the same frequency which may lead high interference. Under HetNets, it may be either neighbour small cell signals (intro-tier cell interference) or local Macro cell signal (cross-tier interference). (Cited from http://4g-lte-world.blogspot.co.uk/2012/06/icic-and-eicic.html)

In the following parts of this chapter, we will introduce candidate techniques that we have adopted in the thesis to solve challenges for HetNets, which include Inter-Cell Interference

Coordination (ICIC), enhanced Inter-Cell Interference coordination (eICIC) and Further enhanced Inter-cell Interference coordination (FeICIC). Meanwhile, we also introduce the concept of self-organization (SON) and potential machine learning (ML) algorithms may help to realise SON in HetNets. Finally, Handover along with Markov Chain Process is illustrated. We will introduce these techniques in chronical order - ICIC was first raised in release 8, eICIC in release 10 and FeICIC in release 11.

1.2 Inter-Cell Interference Coordination (ICIC)

As discussed in last part, if we manage to achieve a frequency reuse ratio of 1, we will suffer great interference, which will significantly reduce the QoS and may even directly hinder the service. In other words, there is always a trade-off between frequency reuse and interference. One method of realising a better balance that has attracted significant interest is ICIC, which is introduced in 3GPP Release 8/9 and is defined as a new air interface in the LTE standard [11]. In contrast to a Reuse 1 system, in which all cells use all frequency resources without any restriction, ICIC uses frequency resources in a cooperative way. In order to establish ‘coordination’ among small cells and macrocells, specific control information is transmitted within the LPN network. And ICIC’s control topology is normally Centralized, where central controller will be responsible to all UEs’ resource allocation. Based on the coordination of central controller, various schemes for ICIC have been proposed and developed, three of the major schemes are as follows:

In the first ICIC scheme, neighbouring eNBs use different sets of RBs and frequency reuse is controlled by a static ratio; thus, only a fixed portion of resources can be shared [13]. This scheme may improve cell-edge SINR and is easy to implement; however, the total throughput of small cells may drop because not all RBs are fully utilised.

In the second scheme, the centre users of all eNBs (regardless of whether they are small or macro) are allowed to fully reuse RBs. No two neighbouring edge users, however, can use the same set of RBs at a given time [14].

In the third scheme, all neighbouring eNBs use different power schemes across the spectrum, with RB assignment following the process outlined by the second scheme explained above. For example, an eNB can employ a power boost for cell edge users with specific sets of resources (not used by neighbours) while maintaining low signal power for centre users. This scheme maintains the availability of all RBs [15].

These schemes show that although the methods to restrict frequency distribution may vary, they are all based on fractional frequency reuse (FFR), rather than full reuse. However, some previous studies have indicated that FFR exhibits relatively poor spectral efficiency [16]. Scheme 1 is the simplest method, requiring no further equipment or algorithm for supporting the pre-set threshold. However, it also has the lowest spectral efficiency and may not be able to adapt to mobile LPN networks. Scheme 2 attempts to improve spectral efficiency, segmenting UEs into two parts according to their QoS and allowing centre users to share frequencies instead of simply setting up one threshold. Therefore, in this scheme, centre users obtain the maximum spectral efficiency; however, edge users still face the same issue. Scheme 3 manages to improve edge users' spectral efficiency by boosting their power, thereby allowing their QoS to reach the threshold for frequency reuse. Although spectral efficiency can be improved by scheme 3, this approach raises another problem: the power surge reduces the working time of the UE.

The other problem is organisation [17]. Resource usage must be planned and distributed across a multi-cell network, and this configuration must be performed automatically due to the large scale and high maintenance cost of the network. Furthermore, given increases in capacity requirements, an increasing number of cells may be added to the network. Each time the network expands, it must re-calculate the frequency distribution plan for edge users. If all these calculations and configurations are processed by humans, the process will be both inefficient and costly.

In conclusion, ICIC achieves a balance between frequency reuse and interference, although issues of low spectral efficiency and complicated organisation remain. Moreover, the ICIC methods specified in Releases 8 and 9 of 3GPP do not specifically consider HetNets settings and may not be perfectly effective for HetNets [18]. Thus, to realise the full potential of HetNets, an evolved technique—enhanced intercell interference coordination (eICIC)—was developed for Release 10.

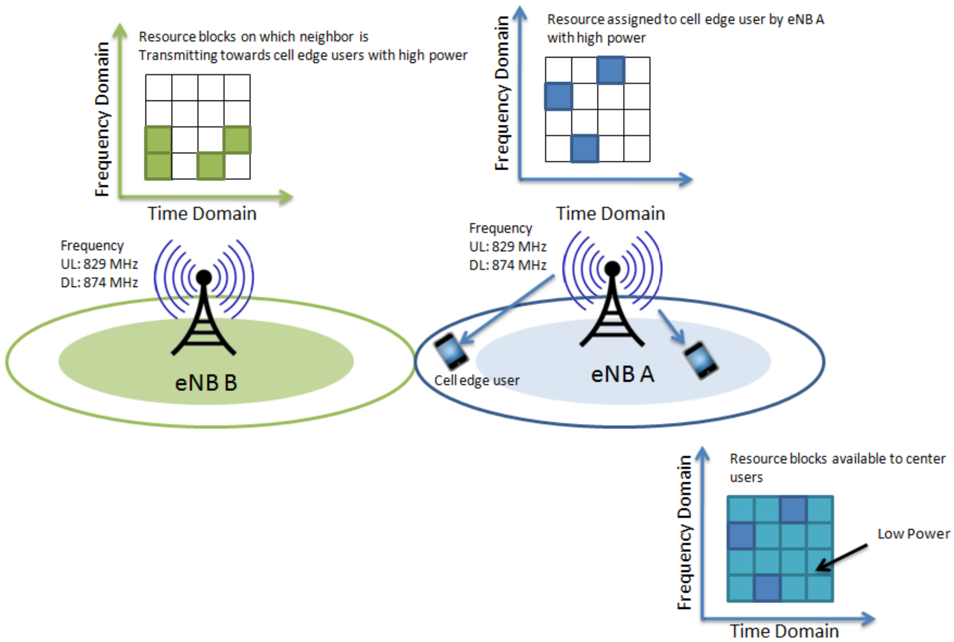

Figure 1-2 ICIC schemes.

No two neighbouring NBs use the same set of RBs at a given time. Meanwhile, RBs assigned to cell edge users will be power-boosted. (cited from: http://4g-lte-world.blogspot.co.uk/2012/06/icic-and-eicic.html)

1.3 Enhance Inter-Cell Interference Coordination (eICIC)

eICIC was first introduced in 3GPP Release 10. Unlike ICIC, eICIC was specially designed to mitigate the inherent inference in HetNets [19]. eICIC can be grouped into three main categories. The first category comprises frequency-domain techniques, where control channels and physical signals (i.e. synchronisation and reference signals) of different cells are scheduled using reduced bandwidths in order to achieve a totally orthogonal transmission of these signals at different cells [20]. The second category comprises time-domain techniques. For the edge users suffering from high interferences, a time delay is introduced to avoid interferences from other-tier nodes. One possible way to achieve this is to use almost-blank subframes (ABSs) at femtocells. ABSs contain no control or data signals. When macrocell and small-cell subframes that contain control or data signals overlap, ABSs can be introduced to break the overlap and mitigate interference [21]. The third category comprises power-control techniques. The desired QoS for macrocell or femtocell UEs can be achieved through the application of different power controls to femtocells (e.g. when the transmission power of a picocell is increased, edge users' total throughput will also be increased) [22]. Technique 1 is more like a modified ICIC; therefore, this part will focus on the other two techniques: power control and ABSs.

1.3.1 Almost Blank Subframe

Although the simple application of power-control techniques such as CRE can partially reduce intra-cell interference by balancing the loads of all small cells, the cross-tier interference originating from the macrocells remains unaddressed. One method to solve this problem is the application of ABSs [10]. To realise this method, we first need to determine the 'victims'—that is, the UEs that suffer from cross-tier interference most severely. Then, macrocells will be muted at specific subframes, allowing the offloaded small-cell users to be scheduled into these blanked time slots. As a result, the interference originating from the macrocells can be eliminated. For example, in [23], the ABS ratio was set to be 0.5, which means that the macrocell must be muted for half of its operating time. Though it seems useless to allow a macrocell to remain idle, during this time, small-cell offloaded users may have greater capacity and better QoS. However, in practice, a muting ratio of 0.5 means that half of the macrocell subframes will be cleared to suit small-cell users. Therefore, there may not be enough non-ABS subframes for all original centre UEs. To address this issue, some previous studies have suggested the use of CRE tools to control the offloading of more UEs to small cells and their allocation to ABSs [24]. This may increase both the load of small cells and the interference of UEs.

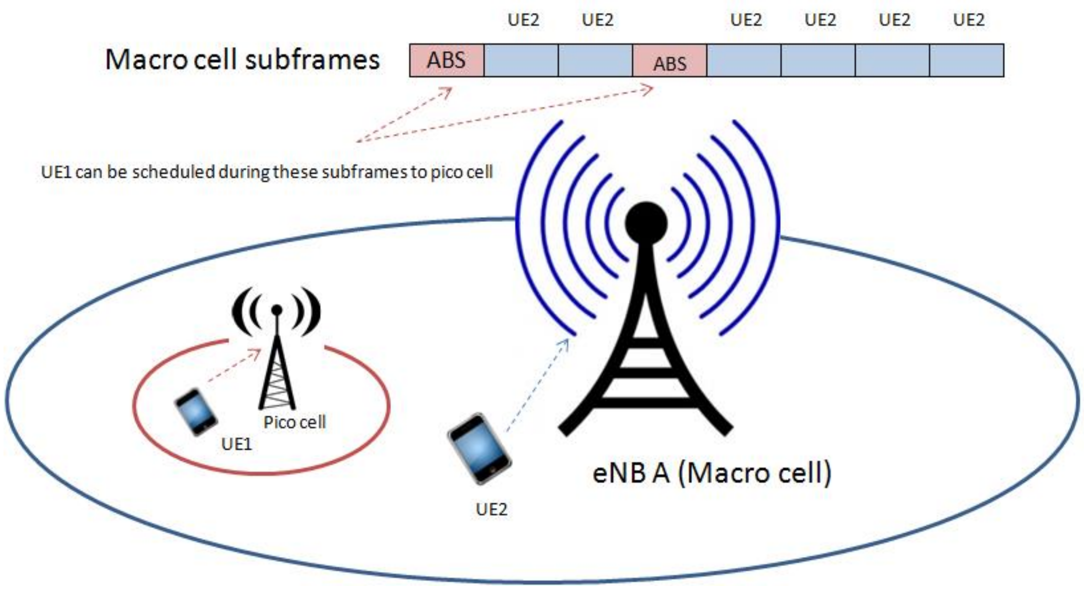

Figure 1-3 Macrocell subframes: ABS and nABS

Macrocell subframes are separated into two types: ABS and normal ones. Central users which suffer from less cross-tier interference will be allocated to normal ones, while the cell edge users offloaded to small cells will be allocated to ABS, when the macrocell is muted. (cited from: http://4g-lte-world.blogspot.co.uk/2012/06/icic-and-eicic.html)

1.3.2 Reduced-Power ABSs (RP-ABSs)

Through the use of ABSs, small cells can use both ABS and nABS subframes. Macrocells, however, are restricted to non-eICIC subframes only, meaning that they are not efficiently utilised, resulting in resource wastage. Therefore, LTE Release 11 implements a new concept, FeICIC, which suggests a modified method: RP-ABSs [25]. With RP-ABSs, macrocells do not completely blank the power on eICIC subframes, and instead use these subframes with reduced power to serve their centre users. This, of course, requires proper self-organisation between the macrocells and the coordinated small cells. The amount of power reduction can be static or dynamic. The capacity gain from using reduced-power subframes over ABSs depends on the ratio of eICIC subframes to non-eICIC subframes within a radio frame, as well as on the intelligence behind the scheduling and coordination. With static RP-ABSs, the ratio of eICIC subframes to non-eICIC subframes is usually fixed, as is the amount of power reduction for all eICIC subframes. However, to obtain a better efficiency when ABS is applied, we can quantise the RP-ABS into several classes (different classes have different reduced powers) so that macro UEs can choose suitable RP-ABS according to their QoS. Obviously, this requires an intelligent coordination algorithm, which may achieve the best utility for the whole HetNets system including not only macro UEs but also small UEs.

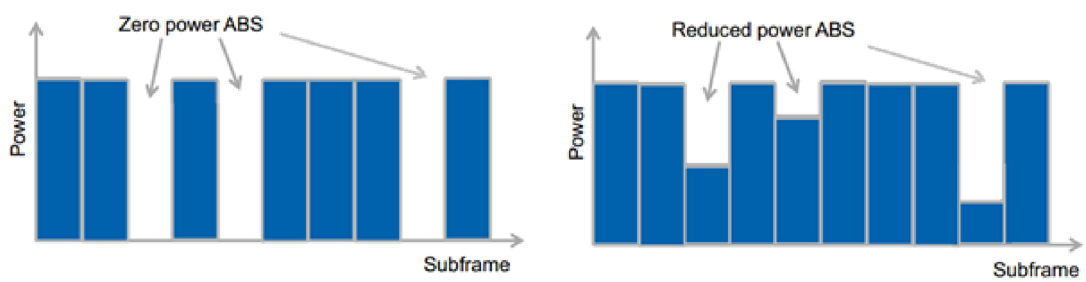

Figure 1-4 Normal ABS (eICIC) and RP-ABS (FeICIC)

(cited from: http://frankrayal.com/2014/05/07/further-enhanced-icic-feicic/)

1.3.3 Cell Range Expansion

In wireless HetNets, due to the differences in transmission power, cell size, and coverage range, even a uniform distribution of user allocation results in an imbalanced load for each single small cell due to 'natural user association metrics' such as SINR, thereby causing intra-tier interferences. Andrews believed that a suitable load balancing scheme would dramatically decrease the interference of HetNets, and suggested the use of the following several algorithms [26]:

Markov Chain processes (MCPs): MCPs, named after Andrey Markov, provide a mathematical framework for studying sequential optimisation in the context of discrete-time stochastic systems, even with uncertainty [27]. MCPs are designed to predict and optimise future work by taking action under the current conditions. Therefore, they may provide a potential approach for implementing self-organising HetNets in centralised network design Moreover, HetNets normally contains large and random traffic map (caused by random mobility model and shadow fading). Therefore, MCP can be applied to establish the framework modelling HetNets Handover process, which inspires us for further analysis later [26].

Game theory: Game theory is a type of mathematical model used to establish cooperation among all possible players or decision-makers (e.g. small cells). It provides 'tractable methods' for organising, even within very large and complicated decentralised networks [27][28]. A game theory algorithm may be suitable for HetNets' decentralised feature. However, the main focus of game theory is strategic decision-making, but there is no closed-form expression to characterise the relation between a performance metric and network parameters. As a result, self-organisation is difficult to be realised in a game theory algorithm, making the network design even more complicated (though it can still be used to develop a general overview and estimation of a network).

Cell range expansion (CRE): CRE, which adds or reduces the bias on the actual received power used to decide user associations, is a well-known power-control technique in 3GPP standardisation [29]. In HetNets, macrocells usually have dominant user control because of their high transmission power compared to small cells (40 W for a typical macrocell and 1 W for a typical picocell) [30]. As a result, most UEs will still attach to a macrocell if traditional user associations are applied, leading to macrocell overload and a uselessness of small cells. In this technique, we define UEs to be allocated to the cells with the highest receiving power. However, we modify UEs' received power from small cells by adding bias, making users more likely to be offloaded to smaller cells. As long as the threshold bias is not reached, UEs will not connect to upper-tier networks. This algorithm significantly restricts the load distribution through a specific expression and is suitable for random decentralised networks. Therefore, it has been very popular in network designs [26].

Given this comparison of existing algorithms, CRE may be the most suitable for network design. However, for real-world applications, several points still require discussion.

Bias values and interference: There are two major ways to consider value biasing: cross-tier deployment and out-of-band biasing [31]. The first method is the original method of using small cellular cells to construct HetNets. The second method is more ambitious: it involves offloading from cellular macrocells to Wi-Fi devices. Of the two, the second approach has a much larger bias—20 dB or more—because it efficiently avoids cross-tier interference. However, due to the high-frequency channel, Wi-Fi coverage is relatively small, such that a normal Wi-Fi access point can only cover 20 m indoors [32]. Therefore, the mainstream study will still focus on finding an optimal cross-tier bias value. The interference arising from CRE is another issue that must be addressed. Cross-tier interference was discussed in an earlier section. Since CRE 'forces' UEs to offload from macrocells to small cells, even if the signals received from the macrocells are higher, the interference signals from other tiers will also be stronger [33]. As a result, although CRE will help organise HetNets, it will also increase cross-tier interference and dramatically drop UEs' QoS.

In conclusion, the comparison shows that CRE is a better algorithm for eICIC implementation, and will be applied in future simulations. Of the two issues remaining, the issue regarding optimal bias value can be addressed through simulation. Interference, however, is more difficult to address because it is generated by the basic concept of CRE—and therefore, cannot be eliminated. In fact, this issue may be even more severe if the CRE value is too high, thus restricting the application of CRE. Therefore, combining CRE and ABSs to provide complementary to each other is one solution to this situation that involves the coordination of macrocells and small cells, and it is discussed in a later simulation section.

In conclusion, the ABS technique is not only capable of mitigating cross-tier interference, but can also be combined with CRE to control the offloading of UEs to small cells to mitigate intra-tier interference. FeICIC further modifies ABS with reduced-power subframes to avoid resource waste.

1.4 Self-Organisation Network (SON)

The previous sections mentioned SON and its importance in realising eICIC. The ABS ratio and reduced power ratio are values we can obtain through simulation and experimentation. SON, however, is more like an abstract concept; thus, it does not have a specific definition. This chapter will briefly discuss SON.

Generally, a self-organising network reduces human work to a minimum in every aspect of the process, from set-up to operation and maintenance [34]. Some previous papers suggest that this minimisation of human work should be performed mostly in the set-up phase [35]. SON provides a promising control system, suggesting that if an algorithm is well planned and considers all possible situations during the set-up phase, then, when the system enters the operation and maintenance phase, set-up is no longer needed and autonomy can be realised. Such a situation requires no human work, even if new devices are added.

Obviously, this is only an ideal and abstract concept. To specifically define SON in HetNets, the following pre-requests should be achieved first [36]:

- Operators' capital expenditure and operating expenditure should be reduced through the minimisation of human involvement.
- No matter how self-organising a network is, capacity, coverage, and QoS should be met first.
- The network should not be fixed, meaning that the network should be capable of foreseeing a high number of new small cells and of handling significant expansion.

And then, SON needs to achieve the following three characteristics. (1) Self-configuration: Given the current operating network, any newly allocated node should be able to set the configuration automatically so that it can adapt to the network without further manual involvement. (2) Self-healing: This feature requires the network to detect system failure and apply the compensation method to fix the issue according to the current network situation. (3) Self-optimisation: The network will automatically find the optimal nodes' deployment and coordination in order to secure the system performance. For a wireless network, the coordination among all cells will be optimised to achieve better QoS, coverage, interference mitigation, etc. The optimisation algorithm may be changing if the optimisation objective is different [4].

Therefore, a closed-loop feedback system is necessary for SON so that the optimisation objective can be maintained in valid range [37]. For example, in simulation, we normally suppose that each end user has the same capacity requirements and an equal resource distribution. In reality, this approach to resource allocation may seriously reduce network efficiency. Different user terminals may have different capacity requirements at different times (e.g. video, audio, and image) [38]. In such a situation, a closed-loop feedback system that can monitor UE requirements in real time and manage capacity accordingly may significantly increase network efficiency. This can also improve the UE's QoS through other means, even when the network and equipment remain the same. Once the closed-loop information is obtained, methods of capacity control, such as the management of transmission power and modulation codes, can be realised.

In the context of HetNets, SON can also contribute to high utility of networks. As mentioned in the last section, CRE is a powerful tool for mitigating the interference caused by HetNets' multiple tiers and the bias value is a crucial issue in the application of CRE. Some papers have applied fixed CRE bias values (e.g. 9 or 12 dB) to all small cells in their simulations [39][40]. This method may reduce simulation complexity, but ignores the particularity of each small cell, including the cell's location, number of surrounding UEs, distance from the macrocell, and transmission power. All these conditions may affect the CRE value. Therefore, applying the same CRE value to all small cells may result in an offloading imbalance, which may cause some small cells to be overloaded, while others to face insufficient UE. SON can help to solve this issue by adding a closed-loop feedback system to monitor the loading situation, reducing bias if a small cell is overloading and increasing bias if a small cell is under-loading. In the end, each small cell will have its own, exclusive CRE bias value, and the system will be balanced.

In addition to helping to balance of CRE values, SON can also benefit HetNets via its third standard. At present, traditional cellular networks are highly complex, with many parameters requiring tuning. These networks are designed primarily for the single technologies involved, and are difficult to reconfigure or design. Therefore, two issues are raised. First, given the dramatic increases in capacity requirements, an increasing number of BSs is required. Due to the high complexity of traditional networks, every time a new BS is added, significant human work is needed, resulting in high costs and lengthy set-up times. Second, with new technological developments, such as LTE, more than one technology is involved in each network; thus, the traditional network approach is no longer valid [41]. SON, however, may avoid such issues. By transitioning parameters from neighbouring nodes, newly added nodes can rapidly self-configure their own parameters, thereby minimising the human work and set-up times. Given the low complexity of such networks, when new technologies are introduced, old-technology nodes can be easily replaced with new-technology nodes. Furthermore, protocols and parameters can be easily reconfigured throughout the network. Thus, such a network is more suitable for the new generation of technology.

One of the most promising and attractive aspects of SON networks is that end users can set up their own BSs at the network terminal, also known as the home BS or femtocell access point [36]. As a result, a new network tier below the macrocell and small cell tiers becomes possible. Owing to SON, these devices can integrate themselves into the existing networks (parameters from neighbouring nodes) without the help of operators. At this time, of course, such UE may no longer be restricted to phones or PCs; other systems, including fire alarms, security systems, real-time traffic monitors, weather reports, and an unlimited number of other options, are possible usage contexts. Therefore, SON may help to realise HetNets in many ways: the balancing of CRE bias values and self-configuration for new cells added to the network. And what we need next is an intelligent technique to enable SON in HetNets.

1.5 Machine Learning Algorithms

ML enables a system to SON and self-building by analysing data. It applies algorithms that allow the system to modify itself without being ‘explicitly programmed’ [42]. In general, ML can be classified into two categories: unsupervised learning and supervised learning.

1.5.1 Unsupervised Learning

Generally, the main objective of supervised learning is to establish a model from the training data with labels. Unlike supervised learning, there is no implication of the pattern or ‘correct answer’ for the input and the output is not provided. The objective of this learning is to find an input dataset pattern in which certain datasets follow more often than the others [43].

One of the highly common-used algorithms for unsupervised learning is **clustering**, which aims to cluster the input datasets into several subsets, in which the elements may generally follow the same pattern [44]. According to the parameters given by the dataset, the algorithm will automatically distribute data elements into groups and each group has similar elements (this similarity is basically defined by their parameters). Once clustering is finished, we can study the pattern of each group and make decisions according to the result, such as taking measures to mitigate negative parameters or finding outliers that are not suitable for this group [45].

The application of clustering is distributed among various fields. Short-range weather prediction can be realised through collecting and clustering daily weather conditions as the database [46]. In the financial field, clustering can be applied to analyse stock prices and find a potential manipulation factor [47]. Image compression uses clustering to group image pixels according to their RGB value (the parameters) so that the pixels with similar colours or patterns are assembled for a better compression [48]. The clustering algorithm is also widely applied in Geography. Through clustering referent vectors of the self-organising map, the model can be used to analyse and measure the colour of the ocean [49].

1.5.2 Supervised Learning

To implement a supervised learning algorithm, the following two requirements should be satisfied: 1) the target and predictor variables should be clarified and listed and 2) sufficient samples implying the 'correct' values for the target variables should be given. The algorithm will learn from these given data to analyse the pattern between the input variables and output results. Therefore, a common supervised learning model will follow a similar methodology to implement and analyse the algorithm [50]:

The first step is to collect the *training dataset.* This set should contain pre-defined values of the parameters and output result variables, for example, a list of patients with the name of their illness as the result variables. Meanwhile, each patient is attached with their gender, age, and occupation as pre-defined values of the parameters. Normally, this training set is incomplete because we cannot collect all patient data in the world, and most importantly, we cannot collect the data for new patients because in the incidents have not occurred at this time point. The algorithm can only generate the model to find the pattern between the input and output for the given data.

As a result, the following procedure will be the evaluation phase of the generated model. For this phase, we need a test dataset, whose characteristic is the same as that of the training dataset. The result variables, however, should be held first for later evaluation. The model generated from the training dataset according to the ML algorithm is then applied to the test data and the predicted result variables are achieved. Next, the predicted result variables are compared with provided ones of the test dataset and the performance of this model can be evaluated. Finally, the model is modified to mitigate the error rate for the given test dataset.

Nevertheless, this modified model may not be satisfactory to predict the unseen data yet, and thus, we need another validation dataset to apply the modified model to it, as done for the test dataset. Further modification will be added to this model till the error rate for the validation dataset is also mitigated to a minimum, and the final version of this model can be applied to predict the unseen data. The whole process is illustrated in Figure 1

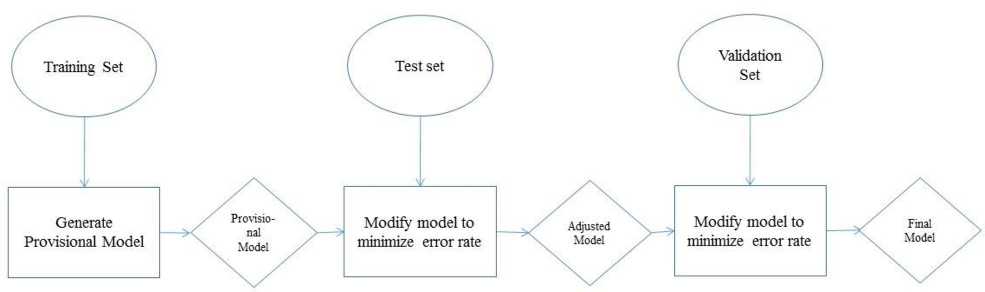

Figure 1-5 Supervised machine learning process

1.5.3 Overfitting vs Underfitting

When we try to evaluate the performance of a supervised ML model, we may introduce a terminology, called ‘fitting’. It is normally used to test the adaptation of a newly built model in statistics [51]. Both overfitting and underfitting will lead to a poor performance of the proposed model.

From the beginning of learning, the error rate of the model will gradually drop as the model continues learning and modifying itself. The model is still in the *underfitting* phase, which requires more relevant feature and more accurate approach to improve. However, if the model includes more features or more complicated approaches than necessary, the noise and random fluctuations of training data will be picked up and caused *overfitting* problem. This will negatively impact model’s performance to new data and therefore reducing model’s ability to generalize [52]. At this time, the error rate will start to increase as the model’s complexity increases. The model has accounted for so much irrelevant information from the training dataset that the importance of useful information is supressed during the computation. Furthermore, the computation time will increase because of the high complexity caused by unnecessary information. Thus, the model becomes a ‘personalised’ version of the training dataset and is not reliable for predicting the result for the test and validation set [52]. As a result, it is very important to understand the background of the data we aim to train so that only relevant features can be included. Our simulations also suggest that direct applying machine learning algorithm without introducing background knowledge will bring poor performance to models.

1.6 Handover

Since we are considering offloading macrocell UEs when small cell network is established, we assume all UEs' initial status is bound to macro cell and handover from currently serving cell to deployed small cell. Handover process usually contains the following two phases: Initiation and Process [53]. During first step, Received Signal Strength (RSS) from serving BS and around candidate ones will be measured and make comparison with pre-defined algorithm. Once the candidate BS is decided and fulfil the requirement of Handover for several TTIs, the process phase will be triggered. During this step, controlling and acknowledge information will be received by UE so that Handover process can be finished according to protocol of new BS. In other words, almost no information signal is transferred during process phase and therefore frequent Handover will lead to significant drop of UE capacity [54]. As a result, the algorithm during initiation phase may be the most important part of Handover. With intelligent designed algorithm, the unnecessary Handover numbers should be mitigated and the computing complex will be reduced to avoid long-time Handover phase. The conventional method only depends on original RSS from serving BS and candidate BS. It defines the process phase is triggered if candidate RSS is larger than serving RSS. Although the computing complexity is minimized, the number of unnecessary Handover will be high because of the fluctuation caused by shadow fading and mobility model [55]. Although the delay of Handover is mitigated, the situation of QoS drop may still be severe.

Alternatively, [56] suggests adding a virtual bias on RSS during the UE association process which represents the threshold of the difference in received signal strength between the serving and the target cells. Besides, Time-to-Trigger (TTT) is introduced, which is the time interval that is required for satisfying HM condition. A handover is initiated if these two conditions are fulfilled: the Receive Signal Strength (RSS) of the candidate cell is greater than the RSS of the serving cell plus the virtual bias and this condition holds at least for the time specified in the TTT parameter. By controlling TTT value, the unnecessary Handover rate can be reduced at the cost of longer Handover delay. In general, if the value of bias and TTT is too high, UE will suffer long Handover delay or even call failure before necessary Handover occurs. Meanwhile, if the value is set too small and may not take effects, the unnecessary Handover will occur frequently causing severe capacity drop.

In order to design an intelligent algorithm to balance the unnecessary Handover rate and Handover delay, we need to understand the cause of unnecessary Handover and the composition of the virtual bias. Ping-Pong handover can be interpreted as a UE handover to a new cell and handed back to the original cell in less than the critical time (Tc), which can result in communication delay, call dropping, capacity reduction [57]. The cause of Ping-Pong Handover may be various but can be generally categories into two source: the situation of UE's location and mobility model of UE [58]. The first source includes weather (rain, snow) and large obstructs like hill or buildings, which may cause significant signal fluctuation. This signal fluctuation at cell edge (normally refers to small cell edge in HetNets) may lead to frequent Handover happens. The second source includes the frequent random movement and UE's high moving speed within the cell edge range, which may cause the decision condition triggered frequently and result in ping-pong Handover. Therefore, these two sources should be considered when we try to solve ping-pong Handover issue.

1.7 Markov Chain Process

So far, we have discussed how eICIC/FeICIC benefit HetNets in solving challenges before and after UE offloading by CRE. However, the effect of CRE during process of handover is hard to define due to complex and frequent ping-pong handover. In order to solve this issue, we need an intelligent model to map this offloading process.

A discrete time Markov Chain $\{M_1,M_2,...M_n\}$ is a Markov stochastic process with countable states space, which changes with process steps T=(1, 2,...) [59]. In other word, we can consider the value of M_n as the result of the nth step process, where the process starts from M_1 (Some paper supposes the state starts from M_0, it is only the difference in later Matrix identifying). Since the state number is a countable and set to be s, we may label the states by $\{S_1, S_2, ... S_s\}$, and we define M_n happens in S_i as $M_n = i$. As a result, suppose M_n is currently in S_i, we can define the probability M_{n+1} happens in S_{j+1} in next step is as follows:

$$P_{ij} = P\{ M_{n+1} = j | M_n = i\} \tag{1.1}$$

In MCP, this is called one- step transition probability, which means the probability for next step only. This formula implies that both the original and final states are involved to affect the probability. If this formula is not involved of the time variable, this MCP has stationary transition probability. However, if the formula value changes as steps changes, in other word, the transit probability will change with time passing by, the MCP will have non-stationary transition probability, which is also the case of our study.

With the definition of transit probability, we can achieve the transit probability between any two states after one step. Suppose i is the row, j is the column, we will be able to establish the matrix of P_{ij}, which is called Markov matrix or transition probability matrix.

$$P = \begin{vmatrix} P_{11} & P_{12} & P_{13} & \cdots & P_{1j} & \cdots \\ P_{21} & P_{22} & P_{23} & \cdots & P_{2j} & \cdots \\ P_{31} & P_{32} & P_{33} & \cdots & P_{3j} & \cdots \\ \vdots & \vdots & \vdots & \ddots & \vdots & \cdots \\ P_{i1} & P_{i2} & P_{i3} & P_{i4} & P_{ij} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{vmatrix}$$

In order to understand the meaning of this matrix, we can consider row i as the probability of all the possible results for next step of $M_n = i$. Since we have assumed that the states number is countable, the transition probability matrix should be a square matrix with the order as the number of states, which equals s [59]. According to the property of MCP and probability matrix, P_{ij} should have the following constraints:

(1) Each element within the matrix should not less than 0, because it represents probability.

$$P_{ij} \geq 0, \qquad \forall\, i, j = 1,2, \dots\, s \tag{1.2}$$

(2) Suppose state number is countable and set to be s, the sum of each row of the matrix should equals to 1. It is because that, the row represents the probability of all possible results (including no state transition, where $j = i$) given $M_n = i$.

$$\sum_{j=1}^{s} P_{ij} = 1 \qquad \forall\, i = 1,2, \dots\, s \tag{1.3}$$

By applying MCP, once the transit probability formula and initial probability distribution of all states M_1 is decided, we can achieve the probability distribution for any step during the process.

According to the formula of conditional probability, we can have following equation:

$$P\{X_1 = i_1, X_2 = i_{2,} \dots, X_n = i_{n,}\} = P\{X_n = i_n \mid X_1 = i_1, X_2 = i_{2,} \dots, X_{n-1} = i_{n-1,}\} \cdot$$

$$P\{X_1 = i_1, X_2 = i_{2,} \dots, X_{n-1} = i_{n-1,}\} \qquad (1.4)$$

According to property of MCP, we can have following equation:

$$P\{X_n = i_n \mid X_1 = i_1, X_2 = i_{2,} \dots, X_{n-1} = i_{n-1}\} = P\{X_n = i_n \mid X_{n-1} = i_{n-1,}\} = P_{i_{n-1} i_n}$$

Then, we can get

$$P\{X_1 = i_1, X_2 = i_{2,} \dots, X_n = i_{n,}\} = P\{X_1 = i_1, X_2 = i_{2,} \dots, X_{n-1} = i_{n-1,}\} \cdot P_{i_{n-1} i_n}$$

$$= P_{i_{n-1} i_n} P_{i_{n-2} i_{n-1}} \dots P_{i_1 i_2} p_{i_1} \qquad (1.5)$$

where p_{i_1} means the probability of M_1 happens in i_1

As a result, with the model of MCP, the probability distribution for any step during the process can be obtained.

1.8 Overview and Contributions

In summary, the innovative topology of HetNets is like two sides of one coin - it can achieve high user capacity and wider coverage range; it will also bring a series of challenging technical issues. The aim of this thesis is to analyse the cause of HetNets challenges, and bring state-of-arts solutions to manage corresponding issues.

The first issue is **interference**, which is also the major issue of HetNets. The reason is that the unique design of HetNets has caused two new types of interferences where traditional single-tier cellular network will not encounter - cross-tire interference and intra-tier interference. The second issue is **offloading**, which is mainly caused by multi-tier cells topology of the network. The difficulty lies not only on technology chosen but also on technology combination, and most importantly, the fairness for whole system UEs. The third issue is Handover, the unbalance power at cell edge may result in frequent Handover. The complex network design also requires Self-Organization. For HetNets, large scale of small cells and high mobility traffic map will keep changing the optimal network setting, which requires the network to have the ability of self-organizing. The main purpose of this thesis is to investigate and propose solutions to be HetNets issues mentioned above. The research work has been overviewed as follows:

Chapter 2

In literature review chapter, we mainly focus on reviewing accomplished work that have been proposed to mitigate HetNets issues. Through critically comparing and analysing existing works, we aim to find out most suitable techniques for our scenario. Three major categories of interference mitigation schemes have been discussed, and we mainly focus on the third one – interference avoidance, which has higher adaption for HetNets scenario. Under this category, dynamic schemes have been suggested to be more suitable than static ones. And then ICIC with centralized control topology, eICIC/FeICIC with semi-distributed control topology, machine learning based SON with autonomous-distributed control topology have been detailed discussed and compared with other related works. As a result, these candidate techniques have shown their advantages among existing works and will be applied to solve HetNets issues in the following chapters.

Chapter 3

In Chapter 3, we have managed to mitigate cross-tier interference of HetNets by combining Almost Blank Subframes and Cell Range Expansion. Through muting macro cell in specific ABS, small cell UEs will benefit from it without cross-tier interference. This chapter firstly apply Nash Bargain Solution with proportional fairness to determine the optimal ABS ratio and UE allocation. Which UE are more vulnerable and how ABS affect small cell UEs are also discussed. With the information from ABS, we propose the Power-Layer Based NBS algorithm to realize reducing power ABS. During Rp-ABS, macro cell power is no longer fully muted, we implement the cost of NBS according to power layer and introduce stepped power reduction, so that both the small cell and macro cell UEs may enjoy a system balance. The optimal Rp-ABS ratio and UE allocation for different layer subframe is obtained and evaluated in the end, which achieves up to 31.4% user throughput gain compared to eICIC and fixed power ratio FeICIC.

Chapter 4

In Chapter 4, we have proposed schemes to solve both offloading problem of HetNets, which includes (1) UE offloading from higher tier to lower tier, and (2) small cell backhaul traffic offloading. This chapter applies a widely used unsupervised Machine Learning (ML) algorithm, K-means Clustering Algorithm (KCA) to address these two offloading issues. For first issue, we propose a User-Based K-means Algorithm (UBKCA) by involving HetNets background and Enhanced Inter Cell Interference Coordination (eICIC) to decide the optimal Cell Range Expansion (CRE) bias given specific offloading objective. The center user group set is established to reduce computing complexity. Meanwhile, CRE bias and Edge User Factor are introduced to enhance user offloading so that loading balance objective can be achieved. Simulations are then performed to show UBKCA's better performance than KCA; the optimal combination of CRE bias and Edge User Factor are taken based on both accuracy and offloading factor; furthermore, we have implemented a close-loop SON system with KNN and linear classification so that new UE will be automatically assigned to suitable network tier and offloading factor is maintained within a moderate range, and cell edge user factor 17.5% and CRE bias of 8dB is optimal combination for our scenario. In order to solve the backhaul traffic offloading, we have managed to establish a cache system within small cell by applying modified KCA. With the help of our proposed cache system, the hit-ratio for our Youtube scenario has been improved to 70.27% and request latency time has been reduced by 60.21%, so that both small cell users' download speed and request time will be enhanced. KNN is then applied to predict new users' content preference and prove our cache system's suitability. Besides that, we have also proposed a system to predict users' content preference even if the collected data is not complete.

Chapter 5

Chapter 3 and chapter 4 manage to solve issue when UEs are in static state. Conversely, chapter 5 has focused on solving ping-pong handover issue during offloading phase within HetNets. Ping-pong Handover can result in communication delay, call dropping, capacity reduction, and this issue may be even more severe in HetNets because of transmission power unbalance. Cell range expansion (CRE), as an important technique of enhanced inter-cell interference coordination (eICIC), can mitigate this issue by adding or reducing the bias on actual received power to enforce user associations; besides, CRE will stabilise UE within specific tier of HetNets and therefore reduce ping-pong handover. However, introducing

CRE will also enhance cross-tier interference and decrease QoS, which makes it quite complicated to determine CRE value. This chapter has applied amodified Markov Chain Process to simulate UE's mobility model and fast fading randomness when UE is trying to Handover. And then use this MCP system to find the optimal CRE value for different kind of scenarios with Markov Chain Process, which is 7dB for our scenario. Furthermore, the difference between CRE and Handover Margin (HM) has been detailed discuss. The combination of CRE and HM is also presented with result. Finally, simulation results will show this proposed method's advantage with other fix CRE value method.

Chapter 2. Literature Review

In last chapter, we have introduced the background of HetNets issues and possible solutions. This chapter will critically discuss the accomplished works on HetNets issue management and find out which techniques may be suitable for our scenario. We will firstly compare existing three major categories of interference mitigation techniques and discuss their capability on solving HetNets issues. Secondly, we will focus on reviewing related works of the third category - **interference avoidance**. In which, the schemes have not only been limited to 'interference' mitigation but also been applied to reallocate resources to solve various network issues, such as UE **offloading, handover, SON** parameters and so on. The schemes are then analysed in terms of resource allocation methods – **statistic or dynamic**. Various network control topologies to realise dynamic resources coordination is then discussed, which includes **centralized, semi-distributed and autonomous-distributed**. The **SON** applications with **Machine Learning** to solve HetNets offloading issue are also reviewed and discussed. Finally, we will summarize potential applicable schemes to solve HetNets challenges after critically reviewing related works.

2.1 Category of Interference Mitigation Techniques

Even before the concept of HetNets is brought up, the contradict of frequency reuse and interference has become a major issue for wireless communication [60]. Although High frequency reuse is one of the key to dramatical increase system capacity, its requirement of sharing same channels among users will also inevitably bring interference and cause serious

QoS problem. Therefore, interference has become the most important issue and is widely studied.

In general, interference mitigation techniques can be categorized as follows [61]: Firstly, *Interference randomization*. The interference is randomized over the whole frequency channels through distributing data transmission on a set of distinct subcarriers so that high frequency reuse can be achieved [62]. Secondly, *Interference cancellation*. This technique is based on spatial filtering so that the interference can be distinguished from normal signal or the signal with best quality will be chosen as serving signal [63]. Thirdly, *Interference avoidance* schemes. This technique manage to modify the parameter of network to reallocate or coordinate resources, this may include both frequency and time domains. Meanwhile, it can also control the transmission power of network cells so that the received signal for users can also be reallocated.

For *interference randomization*, [64] has proposed a method to be able to realize the interference randomization even if UEs have multiplexed in the same Physical Uplink Control Channel (PUCCH). When more than one UE try to multiplex PUCCH to uplink control signal, the difference of index of control channel resource used by any two terminals in the first timeslot of PUCCH is different from the index of control channel resource used by the two terminals in the second timeslot of PUCCH. [65] also introduces a method mapping the control channel elements CCE1-CCEn into a first order of control channel symbol group. After that he produce another groups of symbols with control signal information and then adding these symbol groups with zero values to previous first order control channel symbol group. This combination transfers the first order of the control channel symbol groups into second order, so that cyclic random shifting mechanism can be established and UEs can sharing the same control signal channels. These applications of interference randomization imply that interference randomization mainly focus on solving interference issue on control channel, and is normally applied for UEs' acknowledge message and control signal. It is because that it highly depends on the CQI of control signals to distinguish among signals. Interference aims to realize multiplexing multiple UEs' control signals in same control channel instead of identifying the interference signal and serving signal, so that the system frequency reuse factor may reach a high level; but its limitation of applying in data signal has made interference randomization a poor choice for our HetNets scenario.

For *interference cancellation,* it is normally separated into two categories: Parallel Interference Cancellation (PIC) and Successive Interference Cancellation (SIC), although the difference is getting less with the development of new techniques [66]. [67] applies the concept of PIC to detect all UEs at the same time so that the first initial report can be generated in short time. With this report, severe interference can be detected and cancelled, and the detections on multiple UEs are executed in parallel to further investigate interference. Since this process is repeated in parallel with multiple UEs, PIC can also be considered as multi-stage interference cancellation. Since the initial detection only provide large scale interference cancellation, soft interference cancellation in later parallel detections is necessary. [68] applies the concept of SIC to detect only one UE for every stage. The first stage is to investigate the signal with strongest receiving power, the second stage is to investigate the second strongest received signal. The stages continue till all received signal are processed. And then each signal from whole received signal will be reorganized and constructed, so that this signal can be separated from the composite received signal. In the end, all UEs will benefit from this cancellation process, UEs in early stage suffers little interference because of the high received power, UEs in later stage will suffer less interference because high power signals have been cancelled in early stage. The interference cancellation is no longer limited to control signal only because it does not work on physical layer but on composite signal itself. However, there is still a major issue limiting this technique applying on HetNets – high latency. For SIC, the computing complexity and latency is proportional to number of users, furthermore, this latency is severe for users who request real time data transmission. For PIC, although users are detected in parallel, which may reduce the latency, there are still P cancellation stages after detecting phase.

As a result, [69] has proposed a multi-stage SIC model to establish a relax trade-off between two concepts. UEs are separated into several groups, each group is investigated together and then their signals are also cancelled as a group from the composite received signal, other groups are then detected in parallel. The result shows that this algorithm works well for small number of UEs because detecting UEs in groups will directly reduce time of detecting stages. However, the latency is still a serious issue for large scale network because the complexity of parallel cancellation is still proportional to number of UEs, especially each stage is now a group instead of single UE. As a result, modified *interference cancellation* still has limitation when applying to large scale network, such as HetNets.

For *interference avoidance*, the major objective of this interference mitigation concept is to increase the SINR, which enable all system UEs to share same frequency channels no matter how serious the interference is. Interference avoidance is an algorithm to coordinate network resources and reallocate the resources to UEs in frequency, time and power control [70].

Compared to former two techniques, this technique may have following advantages when dealing with HetNets interference management:

1) Compared to interference cancellation, this kind of techniques will not be limited by number of UEs in system. On the contrary, the core concept of interference avoidance is to consider all UEs' interference situation and enhance each single users' QoS; e.g., [71] applies avoidance schemes to partition all system UEs to suitable cells according to their average SINR on different cell clusters, and then candidate frequency reuse factors are applied on these clusters to find optimal system throughput. For any new UE entering this system, the scheme will calibrate its interference situation and assign it to the cluster which benefit the whole system most instead of single UE.

2) Compared to interference randomization, interreference avoidance is applicable for both control channel and data channel, which is important for large scale network coordination (the performance on control channel may be better through smart coordination); e.g. [72] applies interference avoidance technique on typical a 3GPP-compliment heterogenous network and compare the performance with applying interference averaging based on pseudo-random subcarrier allocation, which is the technique of interference randomization. He states that the randomization schemes are simply generating seeds according to physical cell identifiers (PCIs) and cell radio network temporary identifier (C-RNTIs). Instead, he considers the reallocation of PCIs, C-RNTIs and PDCCH resources and proposes an efficient interference-aware scheduling to control transmission power. The simulation results show that the small cell data channel capacity size will be doubled because of lower interference and trade off the number of PDCCHs given a higher small-cell expansion bias. Meanwhile, the small cell control channel capacity will also be enhanced by at least three times for heavy

loaded and frequently used services such as Voice Call by reallocating macro-cell PDCCH resources.

As a result, although three categories of interference mitigation techniques all have their own advantages and special applications, **interference avoidance may be more suitable for our large scale network scenario – HetNets.** In next part, we will detailed introduce various techniques applying interference concept and discuss their capability.

2.2 Review of Interference Avoidance Techniques

2.2.1 Static versus Dynamic

Interference avoidance techniques have been widely researched based on different network resources, which includes the radio frequency, user association, power distribution and so on. Although these schemes are called 'interference' avoidance, they have also been applied to reallocate resources to solve various network issues, such as UE **offloading, handover, SON** parameters and so on. Therefore, techniques under this category should be classified according to whether the network resources allocation is **static** or **dynamic**. Furthermore, whether the resource (cell, time, frequency, power and so on) is reallocated or coordinated is also detailed discussed by various papers. In order to clearly define how interference avoidance work on network, we adopt the idea of [73] and classify the techniques into two groups according to their methods interacting with network resources:

Fixed Resource Allocation (FRA), which is also called *static channel allocation* in some papers. The major concept of these techniques is to pre-allocate part of network resources (frequency, time, power and so on) to each single cell permanently for their exclusive use. These resources can be evenly distributed among all cells or specifically distributed to necessary cells according to their traffic loads or positions, so that some pre-allocated cells are able to share network resources from neighbour cells.

Dynamic Resource Allocation (DRA): Unlike FRA, there is no pre-allocated resources for cells. All network resources are reserved in the central pool and are distributed dynamically to system cells according to real time network requirement, and then the resources are taken back to the central pool waiting for new coordination. The major concept of DRA is to coordinate the network resources distribution so that the optimal system balance can be reached given certain level of interference constraints.

2.2.2 Fixed Resource Allocation (FRA) Techniques

For FRA techniques, the most important resources that need to be pre-allocated is frequency because the control of frequency reuse is the fundamental concept of FRA [74]. As mentioned in last part, FRA proposes that each cell has its own exclusive part of frequency resource and will not share with its neighbour cell. As a result, due to different methods of combining frequency resource distribution and interference avoidance, frequency reuse schemes contains conventional frequency planning schemes (Reuse-1 and Reuse-3), partial frequency reuse (PFR), and soft frequency reuse (SFR). We will then introduce existing application with these frequency reuse schemes in FRA and discuss their capability for our scenario [61].

Conventional Frequency Schemes

The optimal situation is applying frequency reuse factor of 1 (Reuse-1), which means all available frequency resources are shared and reused among network cells. However, this scheme will inevitably generate severe interference, especially for cell edge users with low received power. For applying FRA, trade-off must be made to reach Reuse-1. In order to directly reduce the interference arisen from reuse-1, [73] and [75] have discussed an replace scheme (Reuse-3) to separate whole frequency resources into three equal but orthogonal sub-bands, so that adjacent sectors can e allocated with different sub-bands and will not interfere neighbouring sectors. The results show that the interference situation can be dramatically mitigated, but the disadvantage is also obvious: the spectrum efficiency is traded off for this strict interference avoidance scheme. As a result, two thirds of frequency resources are not fully utilized compared to Reuse-1.

Partial Frequency Reuse (PFR)

The method of Reuse-3 is quite straightforward by trading off spectrum efficiency and is a basic static resource allocation technique. In order to mitigate the disadvantage of traditional planning, PFR schemes have been applied to increase the frequency factor from Reuse-3. As discussed in last chapter, the common idea of this scheme is to reserve same band and same power level among all sectors. In order to create a low level inter-cell interference for all UEs within the cell, [76] suggests applying two different frequency reuse factors. Full frequency reuse is applied for cell centre users and low frequency reuse is applied for cell edge users for lower interference. [77] modifies the idea by restricting part of the frequency resources from using in some sectors at all. Besides, the reuse factors for cell edge users should be different for each single cell, which is defined according to actual interference situation. [78] modifies the idea of PFR through defining adaptive spectral sharing per cell load conditions. This configuration considers the practical fact that the traffic load in different cells will vary in both contents and request timing. This situation is similar to graph colouring problem and therefore authors applies graph colouring algorithm to solve the resource allocation problem.

[79] proposes a numerical method based on static ICIC to calculate inter-cell interference for cells sharing same frequency channels, and apply coordination according to numerical results. This ICIC technique focuses on mitigating cell-edge UEs interference. It analyse the interference level of these UEs before deciding the portion of reserved frequency reuse, instead of simply setting a universal reuse factor. And then, author compares the inter-cell interference in three schemes, which is uniform frequency reuse factor, Reuse-3 and static ICIC. Simulation results suggest that interference level for cell-edge users applying partial frequency reuse ICIC is nearly two times lower than those applying Reuse-3 schemes, and almost three times lower than those applying universal frequency reuse case.

However, the disadvantage of PFR is also obvious. PFR proposes the policy of strict no-sharing rule for reserved part of frequency resources and therefore is also called Fractional Frequency Reuse with full isolation. As a result, the spectrum efficiency is still underutilization for large scale network although ICIC with PFR has shown superiority over conventional Reuse-3 schemes.

Soft Frequency Reuse (SFR)

In order to raise the spectrum efficiency, some papers propose introducing flexibility to the strict no-sharing rule of PFR, which is also called Soft Frequency Reuse (SFR). Therefore, the word 'soft' means that the frequency reuse can be realized by adjusting *power control* schemes between center and edge bands. [80] and [81] raised the frequency reuse factor for center UEs to 1 so that these users may enjoy the full frequency resources. Authors then use power control to maintain low level interference for cell UEs. The required reuse factor is applied to distribute transmission power to UEs: the center UEs group is assigned with high power because their high interference level of applying Reuse-1, while edge UEs will be assigned with low power so that the total transmission power keeps stable. [82] adopts the idea of interference awareness and calculate the SINR for active UEs. Author divides the groups according to UEs actual SINR instead of central or edge UEs, and assign high power to high SINR group. However, ICIC's idea of SFR may be different to conventional schemes. [83] proposes higher power should be assigned to cell edge UEs because these UEs will suffer more severe inter cell interference. Center UEs still have higher frequency reuse factor and have access to cell edge UEs frequency band but the assigned power will be lower to maintain stable total transmission power. Compared to traditional schemes, ICIC focuses on achieving total UE interference balance within the cell and neighbouring ones while traditional ones only focus on center UEs of current cell with high frequency reuse factors.

In conclusion, FRA techniques does not aim to achieve Reuse-1 for all system UEs. By reserving part of frequency resource, FRA is able to significantly improve certain UEs reuse factor up to 1. Among existing techniques, ICIC may achieve better results by considering actual cell-edge interference situation and neighbouring cell interference level balance. However, this concept also requires ICIC's parameters to be intelligently recalibrated and optimized whenever the situation of cell edge UEs is changed. Furthermore, the study of [84] implies the performance of cell edge UEs may be more sensitive to the frequency resources due to their low received power. The situation will be more severe for small cell UEs in HetNets. Therefore, the reserving frequency resource schemes of FRA may not be applicable for our scenario.

2.2.3 Dynamic Resource Allocation (DRA) Techniques

After discussing FRA techniques, we can find out that the term ‘fixed’ or ‘static’ implies two aspects: 1) the resources are pre-allocated and will not reflect to real-time interference situation. 2) the resources allocation are mostly decided according to current cell situation only. Even if for techniques such as ICIC which analyse cell-edge UEs interference situation from neighbouring cell, they will merely adjust the resource allocation schemes passively within current cell (power control). FRA also assumes the homogeneous cell transmission power and traffic map so that the whole system cell planning is simply ignored. However, HetNets has required the ‘static’ evolving to ‘dynamic’ as well as higher frequency reuse factor, because of its various applications, wide coverage range, complicated connections among tiers and so on. Under this situation, *cell coordination* has become the major technique to achieve DRA techniques. Coordination schemes will allocate resources based on whole network planning. Depending on the topology of controlling, the schemes can be categorized as: **centralized, semi-distributed, coordinated-distributed, or autonomous-distributed** [61].

2.2.3.1 Centralized Topology

The term ‘centralized’ means the network is coordinated by a central controller. This controller will analyse the interference situation of all network UEs and then distribute available Resource Block to these UEs. In order to achieve requested network parameters, such as total throughput, UE fairness and interference level, central controller must receive all Channel Quality Indicator (CQI) from network cells (or eNBs) and send back coordination signals. [85] proposes that the coordination on interference avoidance and dynamic load balancing should be managed through centralized topology, meanwhile coordinated cells are clustered according to location. Author supposes that complete CQI information from all cells and coordination signal back to cells will not be interfered and high total throughput gain will be achieved through prompt real-time traffic offloading. [86] establishes a dynamic scenario where UEs are assigned with mobility models and random data transfers, and then analyse the capacity gains from load balancing with centralized ICIC schemes. [87] proposes a modified algorithm to mitigate the delay issue generated from centralized topology. Author first applies centralized coordination topology to allocate RBs to each cell to achieve maximum total throughput. And then author proposes distributed algorithm to realize ICIC power control among neighbouring cells so that local fine tuning can reduce central controller coordination signalling. However, most of these papers only focus on the advantage by installing a high speed operation central controller in the network, but ignores the huge extra backhaul signalling due to coordination information among central controllers and all eNBs. Even if [87] has suggested a modified topology to reduce signalling, it still does not provide a fundamental method to solve this backhaul issue. This issue may be even severe for HetNets because of the large scale of small cell network.

2.2.3.2 Semi-Distributed Topology

Centralized cell coordination schemes will inevitably generate huge backhaul signalling, this is also the issue for HetNets as discussed in last chapter. As a result, semi-distributed schemes have been introduced to mitigate this problem. Unlike centralized topology, semi-distributed one normally separates algorithm into two parts: the first part is central based, which is similar to centralized schemes and install the central controller to coordinate network eNBs.

The second part is eNB based and central controller only assign resources to eNBs instead of directly assign RBs to UEs. As a result, eNBs will now be responsible to control their own UEs on *frame* level [88].

Conventional Schemes

Therefore, the coordination of RB allocation among eNBs is the major objective of semi-distributed schemes, especially for high frequency reuse factor network. [89] states each eNB should be able to detect which surrounding cell is the dominant source of interference. All UEs will report top two interference signals from neighbouring eNB back to the serving eNB, so that each eNB can establish an interference group according to collected information. In the second step, each eNB will generate a list of preferred RBs which hopes to be banned in neighbouring eNBs. This list will be backhauled to central controllers. And then controller will collect the preferred lists from the whole network and remake a universal list to coordinate all eNBs. This is also the fundamental model of semi-distributed schemes. In order to further reduce overhead signalling, [90] proposes giving eNBs more power to decide RB allocation. With conventional two-level control system, central controller now only give 'suggestion' to local eNBs, while the eNBs have the right to modify the decision according to real time traffic load. [91] transferred RB allocation problem into a fractional graph colouring problem. Author use two-level system to collect the information from all eNBs according to CQI and maps a global interference graph. In this graph, the vertices represent the UEs, and the edges represent critical interference relations between them. The goal is to find a set of colours (set of RBs) for such that there is no conflict between any combinations of colours in the sets.

This two-level cell coordination model has been widely applied and effectively allocated RBs with lower overhead signalling. However, these conventional schemes have assumed that all RBs are allocated with an equal power to simplify the interference analysing in central controller level. Furthermore, these schemes only passively allocate RBs based on the given traffic load map and ignore the situation of overloading or underloading. These two assumptions clearly cannot adapt the fast evolving wireless network in various applications, complicated connections among tiers, wide coverage range and so on. Under such situation, an innovate scheme **eICIC** have been discussed and applied by papers.

eICIC/FeICIC schemes

The high reuse factor (up to 1) spectrum sharing between high power macrocells and low power small cells is the fundamental of HetNets (this is also the reason uniform RB power distribution is no longer valid). In order to realize coordination among different tier of networks, [92] introduces basic idea of eICIC by adopting two-level semi-distributed topology on small cell network. Each macrocell is a central controller and a number of small cells are attached to this macrocell. The macrocell provides coordination and coverage for small cell, while small cells provide cell edge UEs enhanced QoS or cover network black spots. The author also raises two challenges for eICIC: 1) how much radio resources should be traded off to mitigate interference in small cell and 2) new association rules should be defined instead of passively accept the overload situation for macrocell.

As discussed in introduction chapter, eICIC applies Almost Blank Subframes (ABS) to solve the first challenge, which creates protected subframes specially allocate small cell UEs with high cross-tier interference. Meanwhile, CRE is applied to offload UEs from macro cell to small cell to maintain balanced load distribution, which may solve the second challenge. However, the methods to find optimal ratio of ABS and CRE are numerous and sometimes even contradictable. In general, the major conflicts are summarized as below:

1) Total throughput gain versus Cell edge UE throughput gain.

To find a suitable optimization objective is the basic to solve this kind of problems. Some schemes attempt to dynamically find the ABS parameters so that the maximum total throughput is achieved. [93] transfers the ABS resource allocation into a global NP hard non convex optimization problem and his objective is to maximize the total throughput. [94] proposes a joint optimization of power allocation of ABS and resource allocation to maximize the total throughput, so that the optimal ratio of dynamic ABS can be achieved.

To maximize total throughput is a straightforward and effective method for normal optimization problem, but may not be sole dominant factor to decide ABS ratio. ABS is designed to mitigate small cell edge UEs' interference, its effects will vary according to UEs' actual interference situation and even be ignored by center UEs. Simply use total throughput as optimization objective will bring bias for ABS ratio. Therefore, [95] compares the ABS effects on 5 percentile UEs and 50 percentile UEs. Author finds out the effect of ABS has most significant effects on 5 percentile, which is up to 72% throughput gain. The effects fades and reach only 53% throughput gain for 50 percentile small cell UEs. [96] and [97] defines cell edge UEs to be 5 percentile and only considers the gain of this part of UEs to determine ABS parameters.

2) Fixed ratio versus dynamic ratio.

Although eICIC is dynamic resource allocation technique, some papers still argues that the ABS ratio should be fixed to obtain stable small cell interference mitigation. Meanwhile, CRE will be used to dynamically control the number of small cell UEs and maintain the efficiency of ABS. [98] applies fixed 1/10 ABS ratio which means one of ten subframes Macrocell will be muted and small cell will transmit under uniform power over all subframes. And then author applies different CRE bias under 0, 6 and 12 dB to control the UEs allocated to small cell. The result shows under CRE =12 dB, system will achieve best average UE throughput gain with 23% compared to system without ABS. [99] modifies the fixed ABS ratio by pre-defining 4 muting levels – 1/8, 2/8, 3/8 and 4/8, so that the system has certain capability to adapt to the traffic map instead of sticking to one value. The authors suggests that the choice of muting ratio is controlled by CRE value. Higher CRE value will offload more UEs to small cell and therefore rise more interference issues for cell edge UEs, and higher ABS ratio will be introduced to mitigate the situation. Results show that for CRE up to 30 dB, 4/8 will be applied and muting ratio more than 50% is not suggested.

Dynamically controlling CRE value to adapt the system to real-time traffic map may be an option if we assume the interference situation for each UE is the same. However, due to complex topology of HetNets and low power situation of small cell, the actual interference levels of small cell UEs are various and highly sensitive to traffic map changing. Moreover, if CRE value is higher than reasonable value (like 30 dB), it will affect central UE of macro cell and bring negative effects for system. [100] applies a distributed dynamic ABS ratio scheme which assign each macro cell with a changing ABS ratio. The algorithm focuses on optimizing cell edge UEs instead of total UE throughput, and part of macro cell UE throughput are allowed to be sacrificed to benefit small cell edge UEs. Meanwhile, CRE value is fixed to assess the capability of dynamic ABS ratio in coordinating changing interference situation. The results show the algorithm by dynamic ABS ratio has achieved

55.84% gain for 5th tier UEs, where the optimal gain with static ABS ratio is only 46.03%. Unlike CRE controlling, author has considered all real-time cell edge UEs' interference value and used them for calculating optimal ABS ratio for current traffic map. [101] - [103] has further modifies the schemes by introducing game theory to calculating fairness index for each single UE instead of simply calculating throughput. During these papers, Nash bargaining solutions have been applied and become popular for its capability to solving multi-players fairness problems. As a result, dynamically controlling ABS ratio to adapt to interference situation changing and applying CRE to offload UEs only may be optimal schemes for our scenario.

3) Partial fairness versus fairness with cost

We have mentioned in last section that game theory with fairness has been introduced to coordinate ABS subframes resources. In order to transfer our scenario into multi-player game model, the performance and cost for each player (which is UE) should be predefined. [104] adopts the idea of partial fairness (or proportional fairness) in game theory, and ignores the cost of players and only consider their performance. The performance of each UE is defined as its downlink throughput with current ABS schemes. The optimal ABS ratio is then calculated to achieve highest system utility. [105] and [106] modifies the definition of performance from single UE throughput to aggregate cell throughput to simplify the multiplayer scenario. Meanwhile, they also adopt the idea of partial fairness and define the cost of aggregate cell to play this 'game' as zero. Partial fairness is normally applied to simplify multi-player model, however, 0 cost means that UE can switch among players without penalty (or capacity lost for system). Furthermore. This scheme may not suit for FeICIC, where macrocell UEs will also compete to get better resource and 0 cost condition is not adaptable for complex NBS. Therefore, partial fairness may not be suitable for our scenario and reasonable cost for NBS multi-player model is necessary, especially for FeICIC.

4) CRE's effect on handover

To decide the value of CRE is always a dilemma because of its multi effects on HetNets. [107] states that CRE may help to expand small cell range virtually, so that coverage, cell-edge throughput, and overall network throughput are improved. [108] also suggests that CRE has changed conventional user association rule by adding virtual bias on received small cell power, so that UEs may be 'forced' to offload to small cells. And then total throughput will be enhanced due to high spectrum efficiency of HetNets. However, [109] argues that CRE's major objective is to offload UEs and solve load unbalance issue of HetNets. It is not designed to increase UE throughput, on the contrary, too much CRE bias may even bring serious cross-tier interference for cell edge UEs. As a result, CRE will bring negative effects on total throughput.

[110]'s result shows that although positive CRE bias will help to offload UEs to small cell and therefore increase UE's throughput fairness, negative CRE bias will increase total throughput because cell edge UEs staying in macro cell will suffer less interference compared to them staying in small cell. As a result, CRE bias will impact cell edge UE's QoS in terms of throughput.

On the other hand, CRE may also bring positive effect on UEs' throughput in terms of handover phase as discussed in introduction chapter. [111] and [112] indicate that adding bias during user association phase may mitigate unnecessary handover between neighbouring cells, and this ping-pong handover issue may be more severe in HetNets due to power unbalance. [113] further investigates the effect of CRE on mitigating ping-pong handover from system level simulations, which confirms the positive application of CRE on UE throughput. The results show that introducing CRE bias will restrain handover failure rate under 1% while ping-pong handover rate is also maintained under 1%. However, the results also show that CRE's effect is not linear, and the optimal value should be 6dB. [114] also believes that CRE's effect on handover is not fully investigated. Author simulates UEs' mobility performance by analysing ping-pong handover rate with the effects of CRE. Results show that although CRE has limited effects on intra-tier interference in terms of offloading, its positive effect on mitigating ping-pong handover may still benefit UEs in HetNets.

As a result, the decision of CRE is a mixture of offloading, cross-tier interference and handover aspects. Simply considering its positive or negative aspect may not achieve optimal CRE value and requires an intelligent model to solve this issue.

2.2.3.3 Autonomous-Distributed Schemes with Machine Learning Algorithms

As discussed above, the major difference between central and semi-distributed topology is that the resource allocation may happen within eNB instead of backhaul to central controller. Autonomous-distributed schemes, however, further 'decentralized' the network by reducing not only the central coordination but also coordination among eNBs. Each eNB is highly autonomous and assign channels only based on its own UE information, which further eliminating the overhead signalling among eNBs [115]. The most straightforward way is to complete forbidding coordination among eNBs so that communication among eNBs is no longer required. [116] proposes a FFR scheme for constant bit-rate traffic, which requires no signalling among eNBs. The proposed scheme systematically achieves a frequency reuse efficient for a given user spatial distribution. The scheme divides the bandwidth into a number of sub-bands, each consists of a number of sub-carriers. Each eNB constantly performs a "selfish" optimization of the assignment of its power and UEs to sub-bands with the objective of optimizing its own performance by minimizing its power usage. The disadvantage of this 'zero' communication scheme is obvious – it assumes uniform power distribution for all subframes because there is no interference information from neighbouring eNBs, and therefore no protection measures for cell edge UEs. The situation may be even more serious because small cell edge UEs will suffer higher interference due to low received power. Without specific interference mitigation measures, QoS for these UEs is not acceptable. [117] argues that complete coordination isolation among eNBs may bring negative effects for system load balancing, which is highly important for HetNets due to power difference. The scheme provides an inter-cell interference partial coordination and uses power control to apply load balancing. [105] also argues that interference situation from neighbouring small cell is the dominant factor to decide macro cell ABS ratio. As a result, the key of realizing 'autonomous' HetNets does not rely on cutting the communication among eNBs but on how to enable eNBs '*self-organizing*' (SON) with minimum information transmission.

We have discussed the definition of SON in introduction chapter, and indicate that finding pattern and intelligent algorithm is the fundamental for SON. HetNets has complex tiers of network and various standards of equipment, therefore it is hard to summarize pattern from these huge amount of variables. Considering such situation, unsupervised machine learning may be the capable methods to achieve SON in HetNets. [118] aims to solve the problem of traffic congestion of network. The optimal algorithm to distribute RB among UEs with the merit of self-organizing networks is derived through reinforcement learning by observation and interaction with the network. [119] discuss how machine learning methods help to solve handover problem for high-speed UE bounded to small cell. Author proposes a context-aware mobility management procedure for small cell network. The algorithm combines reinforcement learning and eICIC to increase small cell cover range. Furthermore, author also suggests that machine learning should be applied to predict future traffic map so that SON on small cell cover range can be achieved. Results show that high-speed UEs throughput can be improved by 80% and handover failure probability can be reduced by 1/3 on average. [120] introduces fog networking topology into HetNets. The locations of the fog nodes that are auto-upgraded from small cells are specified by unsupervised soft-clustering machine learning algorithm. And then the proposed approach apply simple, but practical, Voronoi tessellation model to efficiently reduce average system latency. In the end, closed-loop error control system is established to monitor average latency within required range. Compared to complete isolated autonomous system, machine learning based SON system still requires minimum necessary information exchange among neighbouring eNBs, which includes interference situation, cell edge UE locations and ABS arrangement and so on. Although these data transfer may increase overhead signals, they are also vital information for SON system establishment. Review papers show that not only average latency drops, total throughput may also increase especially for cell edge UEs. Moreover, the effects of machine learning based SON may continue to improve as more data is collected and algorithm will keep modifying itself to adapt changing traffic map.

2.2.3.4 Machine Learning Algorithms Comparison on Solving Offloading Issue

We have discussed and reviewed the capability of ML on establishing SON system in last part. In this part, we will review popular machine learning algorithms' applications in wireless communication and find suitable algorithm for one of the possible SON system in HetNets – offloading. Conventional algorithms used in solving offloading issues contains: reinforcement learning, Q-learning and K-means clustering. For reinforcement learning, [121] applies this algorithm to solve power control problems so that optimal energy efficiency can be achieved. Author used Markov Chain Process and define the states as the battery state, the channel state and the packet transmission/reception states. The transit probability is trained by channel state information feedback on UEs' energy efficiency situation. In order to obtain the transit probability formula, flexibility has been introduced by applying the maximum-likelihood heuristic policy and the voting heuristic policy. [122] proposes a coordination-based and context-aware mobility management procedure for small cell networks by applying reinforcement learning. Macro cell and small cell will share long-term traffic map and therefore learn how to obtain optimal cell range expansion based on not only higher UE throughput but also better fairness. The algorithm will keep reinforcing modifying itself with the change of traffic map. Meanwhile, Q-learning is similar to reinforcement learning and also based on MCP. In specific, Q-learning modify the algorithm by introducing an agent, this agent will receive reward if it take actions on a specific state and the goal is to maximize the accumulated reward. And the reward is illustrated by a Q-function, in which "Q" is defined to be an fixed random number [123]. Author of [124] applies Q-leaning to solve both the resource allocation and interference coordination problems in HetNets. Firstly, the algorithm will predict possible available spectrum resources through continue learning from existing RB allocation situation. Secondly, these available resource will be specifically assigned to small cell network so that small cells may attract more UEs (higher reward) to allocate in these RBs. [125] proposes a modified Q-learning model by introduce Hidden Markov Chain design so that Markov chain process will converge to a bounded near optimal distribution even if only part of system information is provided. Moreover, game theory with payoff-based log-linear learning is also combined with Q-leaning so that system fairness is also obtained.

Unlike last two algorithms, K-means partition elements according to clustering instead of making decision based on probability. Clustering problem has been widely applied to solve issues in wireless communications, which includes interference mitigating on Coordinated multi-point transmission (CoMP) [126], devices clustering in D2D networks to achieve high energy efficiency [127] and WiFi users clustering to maintain optimal access point association [128]. Therefore, we may also apply K-means to clustering UEs in HetNets to obtain optimal user association between macro cell and small cell. After reviewing applications of these algorithms on offloading issues, we can find out that conventional reinforcement algorithm focuses on establishing a strict mathematical model based on probability. With exact transit probability formula and complete Markov Chain model, algorithm can accurately map detailed system information which helps to make offloading decision. However, this 'detail' will also limit the algorithm in terms of complexity and calculating time. Therefore, Q-learning has modified the algorithm by introducing Q-function on specific state so that we can train the algorithm even if we don't know the whole model. The problem of Q-learning is also obvious – the performance of the algorithm is highly dependent on how well we define Q-function, this also reduce the adaption of the algorithm to fast changing traffic map. K-means also requires no model establishment, besides, elements will form groups based on similarity instead of function or formula. Although this may reduce the accuracy of decision, it dramatically increase algorithm's calculating speed and adaption to large scale of network, which may be more suitable for our HetNets offloading scenario. Additional techniques should be combined to maintain high accuracy (such as CRE).

2.3 Summary

Within this chapter, we firstly discuss three major categories of network issue mitigation and find out **interference avoidance** may be optimal category for our HetNets scenario. It is because schemes in this category will not be limited by huge size of network and also applicable for both control and information signals. After that, fixed schemes of this category are critically discussed, and three types of **ICIC** schemes are specifically analysed. Although **Soft Frequency Reuse** has shown advantage over conventional fixed interference avoidance schemes by considering cell edge UEs as well, it still cannot reach the requirement of HetNets Reuse-1 standard. As a result, **dynamic** schemes of interference avoidance category may be more suitable for our scenario because three major issues of HetNets can be mitigated through intelligent dynamic resources allocations and self-organisation topology, which includes **semi-distributed (eICIC/FeICIC), autonomous-distributed with machine learning based SON (K-means algorithm and Markov Chain Process).** In the following chapters, we will detailed analyse candidate techniques and propose solutions to HetNets issues.

Chapter 3. Cross-tier Interference Management with eICIC and FeICIC

3.1 Introduction

As discussed in literature review chapter, enhanced Inter-cell Interference coordination (eICIC) and Further Enhanced Inter-cell Interference coordination (FeICIC) may help to solve the first challenge of HetNets, which is the cross-tier interference. In this chapter, we will firstly introduce Nash Bargain Solution (NBS) and transfer the eICIC problem into an N-player NBS problem. The first simulation is to apply partial fairness NBS and implement eICIC. Based on the analysis from case 1, we propose a multi-layer NBS algorithm in case 2 to implement FeICIC. The performance for both simulations are evaluated by comparing with existing eICIC parameters and fixed power ratio FeICIC. In conclusion section, contribution and further work is presented.

3.2 Methodology

3.2.1 Apply Nash Bargain Solution for eICIC

In introduction chapter, we have introduced the idea of eICIC and its application in solving cross-tier interference. In this part, we will detailed discuss our proposed method of ABS implementation. Nash Bargain Solution is a concept of game theory that helps to distribute limited resources among candidate competitors. One simple description of NBS is shown as below: N= {1, 2, …N} competitors require the usage of resources, but resources are not enough to satisfy all of them (obviously, each player's best expectation is to obtain all the resources). In order to obtain reward, everyone will have to demonstrate their performances once they receive the resources, and summarized as set P = {P1, P2, …, Pn}. Meanwhile, all the competitors will also have to pay a cost to earn the right to enter this competition, and summarized as set C = {C1, C2, … Cn}. After considering both their performance and cost, each user will gain his proportion of the resources (some user may even gain nothing on extreme situation if their performance is less than cost). This situation is called N-person bargaining problem, and the formula is shown as below:

$$U = \underset{x}{argmax} \prod_{i=1}^{N}(P_i(x) - C_i(x)) \tag{3.1}$$

For (3.1), the performance $P_i(x)$ and cost $C_i(x)$ of all players (from i=1 to N) are functions of variable x. Our objective is to find this optimal value of x and the corresponding maximal utility (U). Therefore, we can formulate our scenario into a NBS problem and solve it, which means we need to define the players, resources and cost in this formula. For eICIC, ABS can provide better QoS but has limited numbers, small cell UEs will compete to gain ABS instead of normal subframes. Macrocell UEs, meanwhile, can only be allocated in nABS. This situation can be considered as N-person bargaining problem. However, in real simulation part, this computation is quite complex to realize. Firstly, small cell may be bounded by dozens of UEs, and computation complexity may be severe due to large number of N. Secondly, N is not same for different small cells. It may be difficult to formulate a simple common equation to cover all the small cells during simulation part.

Nevertheless, for each small cell under eICIC situation, the type of subframes are certain – either ABS or nABS. Therefore, we can consider nABS and ABS as two players competing to get more subframes. Since small cell UEs will be allocated in these two type subframes, the original N-player problem is transferred into 2-player problem once UE are partitioned (for small cell). In mathematical way, we set nABS to be player 1, and ABS to be player 2. In such case, P = {P1, P2} and C = {C1, C2}, where P1 is the total performance of nABS when his proportion of UEs are allocated. In our scenario, P1 can be considered as the total capacity of nABS UEs of current small cell. Similarly, P2 can be considered as the total capacity of ABS UEs of current small cell. The respect formula is shown as below:

$$P1 = \frac{BT(1-r)}{a_n} \sum_{j=1}^{N} a_j log_2 \left(1 + SINR_{a_j}\right) \tag{3.2}$$

$$P2 = \frac{BTr}{b_n} \sum_{j=1}^{N} b_j log_2 \left(1 + SINR_{b_j}\right) \tag{3.3}$$

$$a_j + b_j = 1 \tag{3.4}$$

In (3.2) and (3.3), the objective of formula is to calculate total throughput of nABS UEs and ABS UEs by applying Shannon Capacity Theorem. In which, j represents which UE is under calculation; B is the bandwidth; a_j is the probability of UE allocating on normal subframes. b_j is the probability of UE allocating on ABSs. Combining these two conditions, each small cell UE must be allocated in either normal ones or ABSs, and secures the condition of 2-player problem (3.4). *r* represents the ratio of ABS, and it equals to number of ABS over total subframes. T is total number of subframes, $T(1-r)$ and Tr can be considered as total transmitting time of normal ones and ABS respectively. a_n and b_n represents total number of UEs allocating in non-ABS and ABS respectively. Left part of formula before $\sum$ means the bandwidth are shared by corresponding UEs, and right part of formula after $\sum$ $\log_2(1 + \mathrm{SINR})$ is part of Shannon Capacity Formula which used to calculate downlink capacity of UE; $\sum$ is used to calculate total capacity of all involved UEs. N is the number of small cell UEs. As a result, P1 and P2 are defined.

So far, we have defined performance for small cells. Since we aim to reach a maximum utility for whole HetNets system, we should also take macro cell UEs into consideration. Unlike small cell, there are no ABS users for macro cell because it is muted during this period, and c_j = 1. Therefore, total capacity for macro cell UEs can be calculated as follows, where the parameters are similar to (3.2):

$$P3 = \frac{BT(1-r)}{c_n} \sum_{j=1}^{N'} c_j \, log_2 \left(1 + SINR_{c_j}\right) \tag{3.5}$$

For cost part, [98] supposes the cost of players can be assumed as 0 in eICIC to obtain the proportional fairness. Since the 'player' ABS or nABS is integrated by UEs, 0 cost means that UE can switch between two players without penalty (or capacity lost for system). For sim1, we have adopted this assumption for simplicity. However, this may not suit for FeICIC, where macrocell UEs will also compete to get better resource and 0 cost condition is not adaptable for complex NBS. We will further discuss the cost in case 2 part.

$$U = arg \max_{r,a_j,b_j} P_3 \prod_{i=1}^{N_s} (P_1(i,r,a_j,b_j) - C_1(i,r,a_j,b_j))(P_2(i,r,a_j,b_j) - C_2(i,r,a_j,b_j)) \tag{3.6}$$

As a result, we can transfer the NBS problem according to HetNets scenario, and our final objective equation can be summarized in (3.6). Firstly, if we consider 2-BS HetNets model, this formula will change 2-player NBS to 3-player NBS by introducing macro cell as well to maintain whole HetNets utility fairness (Ns =1 at this time). Secondly, we have stated that the ABS ratio and UE partition will control the parameters of eICIC and hence affect the capacity of UEs. Therefore, the performance P is a function of r and a_j, b_j separately. And our objective is to find this balanced point of r and a_j, b_j to maximize the total utility. Thirdly, Since P1 and P2 is proportional to the number of bounded UEs, if either one bounded too much UEs, the other one will suffer loss accordingly (total number of UEs is fixed). As a result, total utility U will drop because these two parts are related by multiplication. To obtain optimal system utility, they should cooperate instead of conflict.

After establishing the objective formula, we need to understand how UE reacts to ABS and intelligently partition UEs. (3.2) and (3.3) require the summation of capacity for all UEs within ABS and nABS, which makes the chosen of UE is essential for algorithm. According to Shannon theorem, UE's capacity is proportional to its SINR. Therefore, we align all UEs bounded in one small cell in descending order according to their distance from the cell. The reason of this arrangement is that the distance from cell is one dominant factor affecting UE performance. According to Path loss model, signal strength drops exponential to distance. Cell edge small UEs may be more vulnerable to cross-tier interference. In other words, cell edge UEs (long distance to small cell) may have higher priority to enjoy the benefit of ABS. Meanwhile, according to the definition of [20], the SINR equation for ABS UEs is different from normal small cell one (3.9) by eliminating interference from macro cell (cross tier interference), :

$$SINR = \frac{P_s l_s g_s}{\sum_{i=1}^{N_s} P_i l_i g_i + \sigma^2} \qquad (3.7)$$

$$Path\ Loss = 20\ log_{10}(d) + 20\ log_{10}(f) - 147.55 \qquad (3.8)$$

Ps and *Pi* represent the transmission power of small cell, where *Ps* is the power of serving small cell. l_s *and* l_i represent free path loss, and the expression in dB is (3.8). In which, *d* is distance from serving cell to UE, and *f* is carrier frequency. The constant number of 147.55 is calculated when the unit of *d* is meter and *f* is Hz, and the variation of this number is due to process of dB conversion. The actual number may vary if the unit of *d* and *f* is different (such as Km and GHz). g_s and g_i is the fast-fading gain which is assumed as Rayleigh Distribution. As a result, the expression of *g* should be the exponential random variable. And $P_s l_s g_s$ represents received signal from serving small cell with free path loss propagation model. Therefore, the interference from macrocell is eliminated in (3.7).

As described all above, we have formulated the following algorithm to find optimal ABS ratio *r* and UE partition:

Algorithm 3.1: eICIC design with partial fairness NBS

Input:

- Set constant value B, T to feasible values
- Set $C_1 = 0$, $C_2 = 0$ for partial fairness NBS
- Initialize $r = 0.05$ (the iteration step is also set to 0.05)
- Small cell number N_s, number of UEs in current small cell N
- Set optimal small cell utility $S_i = 0$, and corresponding temporary variable $S_i' = 0$
- Set optimal number of UEs allocating in nABS $k = 1$, and corresponding temporary variable $k' = 1$
- Set optimal total utility $U = 0$, and corresponding temporary variable $U' = 0$
- Set iteration step variable $i = 1$

Initialization and Iterations

Initial $U = 0$
for $r = 0.05$ to 1 (iteration step = 0.05) **do** (loop for possible ABS ratio)
  for $i = 1$ to N_s **do** (loop for all small cells within HetNets)
    Initialize $k' = 1$
    Initialize $S_i' = 0$
    Sort UEs in i_{th} small cell in descending order according to their distance to small cell
    for $k = 1$ to N **do** (loop for all UEs in ith small cell)
      Allocate UE 1 to k into nABS
      Allocate UE $k+1$ to N into ABS
      P_1 = total capacity for nABS users
      P_2 = total capacity for ABS users
      $S_i' = (P_1 - C_1) * (P_2 - C_2)$
      If $S_i' > S_i$ **then**
        $k_i = k_i'$ to replace optimal user partition
        $S_i = S_i'$ to replace optimal utility for current small cell
      end if
    end for
  P_3 = total capacity for macro cell users under current r
  $U' = P_3 \prod_{i=1}^{N_s} S_i$
  if $U' > U$ **then**
    $U = U'$ to replace optimal total utility for whole system
  end if
end for
Return U, k, r

3.2.2 *Power-Layer Based NBS for FeICIC*

FeICIC is a combination of time-domain and power-control techniques. In FeICIC, macrocells' transmission power is reduced during ABS instead of being simply muted. Macrocell UEs prefer to be allocated in nABS for better QoS, which is opposite to the small-cell situation. As a result, competition occurs not only in small cells but also in macrocells for an FeICIC scenario. To implement the FeICIC design, we need to make three essential decisions: distribution of UEs, ratio of ABS, and ratio of reduced power.

To analyse the difference between eICIC and FeICIC, we need to understand how macrocell UE behaves in rp-ABS. We have stated that (3.8) shows the free-space path loss model in dB, where *d* is the distance between the cell and UE and *f* is the carrier frequency. SINR of macro UE can directly affect UEs' capacity, and can be derived in (3.9), where the numerator shows the UE's receive signal strength (RSS) affected by the free-space path loss and shadow fading. The denominator contains two parts: interference part, which is the total signal strength power obtained from other surrounding cells (contain both cross-tier and intra-tier interferences, which is different from 3.8) and thermal noise part (σ^2). Therefore, a UE's SINR is proportional to its RSS from the serving cell. For a macrocell UE, a higher RSS means that the UE is close to the macrocell and low-layer ABS is enough to meet the UE's QoS requirement; lower-RSS UEs, however, require high-level power subframes to maintain moderate SINR while suffering from a large cross-tier interference, and should be allocated to nABS. As a result, we align both macrocell UEs and small-cell UEs in descending order according to their distance from serving cell, to meet various power-layer subframes with decreasing SINR.

A conventional eICIC design sets a universal ABS ratio and fixed power ratio. However, the fixed power ratio may not satisfy all UEs' QoS requirements because of their various locations and SINR situations; besides, the efficiency of FeICIC is not fully exploited by simply setting up one type of ABS. Therefore, we propose a multi-layer ABS according to their power ratio, so that UEs may have options to choose which layer suits their QoS requirement best. In this paper, we design two types of rp-ABSs with power ratios of 0.67 and 0.33. Considering nABS and small-cell subframes, the system comprises six layers and we can formulate this situation as a six-player NBS problem.

$$SINR = \frac{P_s l_s g_s}{P_m l_m g_m + \sum_{i=1}^{N_s} P_i l_i g_i + \sigma^2} \tag{3.9}$$

Let us assume that this UE location is fixed. The first layer is nABS for the macrocell, during which the macro UE may enjoy the best QoS because of unmodified transmission power in the numerator of (3.9). Compared to (3.8), this part represents received power from neighbouring macro cell as interference. P_m is transmission power of neighbouring macro cell, l_m represents path loss from neighbouring macro cell, g_m represents fast fading gain. The rest symbols are the same to (3.8). The second and third layers are rp-ABS with power ratios $b_1 = 0.67$ *and* $b_2 = 0.33$. If UEs are assigned to rp-ABS instead of nABS, their SINR may drop according to (3.9), because their RSS is forced to drop by 33.3% and 66.7%. As the UE is further away from the macrocell and closer to the small cell, it will be offloaded to a small cell through CRE bias. As a result, the UE may suffer from severe cross-tier interference and SINR will drop dramatically. These vulnerable small-cell UEs will be allocated to the fourth layer rp-ABS with $b_2 = 0.33$—so that they can enjoy the best benefit of FeICIC as well as QoS. As the UE gets closer to the small cell, the severity of cross-tier interference reduces because RSS in the numerator gets stronger. Finally, the core small-cell UEs are allocated to the six-layer subframes, nABS, where they receive no benefit from FeICIC. (3.10) – (3.20) show the performance and cost of the six layers (C_6 is 0 because UE in this layer receives no benefit from FeICIC and no worse situation is available).

$$P_1 = \frac{BT(1-a_1-a_2)}{n_1}\sum_{i=1}^{N} b_{1j}\log_2(1+SINR_1) \tag{3.10}$$

$$C_1 = \frac{BT(1-a_1-a_2)}{n_1}\sum_{i=1}^{N} b_{1j}\log_2(1+SINR_2) \tag{3.11}$$

$$P_2 = \frac{BTa_1}{n_2}\sum_{i=1}^{N} b_{2j}\log_2(1+SINR_2) \tag{3.12}$$

$$C_2 = \frac{BTa_1}{n_2}\sum_{i=1}^{N} b_{2j}\log_2(1+SINR_3) \tag{3.13}$$

$$P_3 = \frac{BTa_2}{n_3}\sum_{i=1}^{N} b_{3j}\log_2(1+SINR_3) \tag{3.14}$$

$$C_3 = \frac{BTa_2}{n_3}\sum_{i=1}^{N'} b_{3j}\log_2(1+SINR_4) \tag{3.15}$$

$$P_4 = \frac{BTa_2}{n_4}\sum_{i=1}^{N'} b_{4j}\log_2(1+SINR_4) \tag{3.16}$$

$$C_4 = \frac{BTa_2}{n_4}\sum_{i=1}^{N'} b_{4j}\log_2(1+SINR_5) \tag{3.17}$$

$$P_5 = \frac{BTa_1}{n_5}\sum_{i=1}^{N'} b_{5j}\log_2(1+SINR_5) \tag{3.18}$$

$$C_5 = \frac{BTa_1}{n_5} \sum_{i=1}^{N'} b_{5j} \log_2(1 + SINR_6) \tag{3.19}$$

$$P_6 = \frac{BT(1 - a_1 - a_2)}{n_6} \sum_{i=1}^{N'} b_{6j} \log_2(1 + SINR_6) \tag{3.20}$$

$$U = (P_1 - C_1)(P_2 - C_2)(P_3 - C_3) \prod_{i=1}^{N_s} (P_4 - C_4)(P_5 - C_5) P_6 \tag{3.21}$$

where B indicates the bandwidth, a_1 and a_2 indicate the percentages of total subframes for two types of rp-ABSs, N and N' indicate the total UE number in the respective macrocell or small cell, $b_{1j} - b_{6j}$ indicates whether the current UE is in this layer (can only be 0 or 1), and $n_1 - n_6$ represents the number of UEs in this layer.

Now, we need to detailed define the cost for each layer (or player in game theory) to formulate this multi-player NBS system. As discussed in last section, partial fairness which sets cost of each player as 0 is not practical for FeICIC design because of the increasing type of ABS in both cells. If we still set cost of entering a layer to be 0, UE will be much easier to shift layer and the top layer of macrocell will have huge advantage to attract UEs because of the high transmission power, which may break the balance of NBS. As a result, we decide to set the cost of each layer with opportunity cost. The first 3 layers in macro cell represents three different situations of SINR. The first layer UE may enjoy best SINR boost because they are allocated in ABS and enjoy full transmission power. However, if this UE want to enter this layer, he must give up his SINR boost when he is in second layer, and enjoy 67% of full transmission power, which become the opportunity cost of it. The difference between (3.10) and (3.11), will be the gain when UE is in layer 1. In general, the six layers are aligned according to their ability to maintain UE's QoS. If UE wants to shift to a higher layer and enjoy better SINR, it will have to give up his performance in current layer. In other words, the performance of a UE in lower layer is this UE's opportunity cost, the gain of shifting layer is just his contribution to enter NBS. The benefit of this dynamic step layer structure is in 2 fold. Firstly, this layer structure may exploit the potential of every single UE according to its SINR situation instead of evenly considered in fix power ratio structure. Secondly, if one player in NBS has a negative gain, this player will be refused to enter the system. In our scenario, it means that this UE will not contribute to total capacity and cause the waste of resource. The layer structure ensures UE will eventually find its position that bring best gain no matter in macro cell or small cell, so that no negative gain may happen. After defining the performance and cost of 6 players, we can calculate the total utility according to (3.16). By iterating the combination of a_1 and a_2 and optimal UE partition from algorithm 1, we will be able to find optimal ABS ratio for both type of rp-ABS, which maximize (3.16). The algorithm shows the whole process is shown below:

Algorithm 3.2: FeICIC with Power-Layer Based NBS

Input:

- Set constant value B, T to feasible values
- Set iteration step variable i =1
- Set maximum ratio allowed for ABS, L to feasible values
- Set L as maximum rp-ABS rate
- Initialize two rp-ABS ratios a_1 = 0.01, a_2 = 0.01 for six-layer model
- Small cell number N_s
- Set optimal total utility U = 0, and corresponding temporary variable U' = 0

Initialization and Iterations

1. Initial U = 0
2. Sort macro UE in descending order according to their distance to small cell
3. **for** a_1 = 0.01 to L - 1 (iteration step = 0.01) **do**
4.   **for** $a_2 = 0.01$ to $L - a_1$ **do**
5.   Initialize U' = 0

6. P_1 = gain of macro cell when the first part of UE is calculated according to (3.10)
7. C_1 = cost for macro cell when the first part of UE is calculated according to (3.11)
8. P_2, C_2, P_3, C_3 follow (3.12) – (3.15) for second and third part of macro UE
9. **for** i = 1 to N_s **do** (loop for all small cells)
10. Sort small UEs in descending order according to the distance to i_{th} small cell
11. Partition UEs in each cell according to Algorithm 1
12. P_4 = gain for current small cell when the first part of UE is calculated according to (3.16)
13. C_4 = cost for current small cell when the first part of UE is calculated according to (3.17)
14. P_5, C_5, P_6, follow the (3.18) – (3.20) for second and third part of small UE
15. **end for**
16. $U' = (P_1 - C_1)(P_1 - C_1)(P3 - C3)\prod_{i=1}^{N_s}(P4 - C4)(P5 - C5)P6$
25. **if** $U' > U$ **then**
26. $U = U'$ to replace optimal total utility for whole system
27. Record current a_1 and a_2
17. **end if**
18. **end for**
19. **end for**
20. Return U and corresponding a_1, a_2

3.3 Performance Evaluation

3.3.1 Simulation of eICIC: Case 1

Theories and algorithms of the proposed eICIC schemes have been introduced; we will then show our simulation results and take discussions. This part will analyse the results in three aspects:

- First one is to find optimal ABS ratio with proposed algorithms, and then evaluate the outcome;
- Second one is to test the benefit of eICIC compared to normal HetNets and find out how eICIC affects small cell UEs;
- Third one is to compare our proposed schemes with existing schemes.

3.3.2 System Model

For this simulation, a two-tier HetNets system is built up with one central macrocell of circle coverage for simplicity, and the radius is 500 meters. 4 small cells are allocated in four quarters of the circle with 300 meters away from macrocell. 500 UEs are randomly distributed within the macrocell coverage circle. CRE = 9dB is set to offload UEs. HetNets frequency reuse scheme is set to Reuse-1. Therefore, subframes share the same frequency resources but separated by time and each subframe is 1ms in time. Actual simulation parameters are shown in table 3.1.

Table 3-1 Important simulation parameters for Simulation 1

Parameters	**Value**
Bandwidth	*10 MHz*
Cell layout	*Two-Tier HetNets*
User Equipment Number	*500*
Transmit power of macro cell	*40W/46dBm*
Transmit power of small cell	*0.25 W/24 dBm*
Noise power	*-174dBm*
CRE	*9 dB*
Subframe number	*1000*
ABS ratio (summary of both types)	*1%-100%*

3.3.3 Simulation result and analysis

Figure 3.3 shows total utility changes with ABS ratio r. It indicates that *U* starts to increase from $r = 0$ (which means no eICIC is applied) and reaches maximum when $r = 0.42$; after that, *U* gradually declines and reaches minimum when $r = 1$ (which means macro cell is totally muted). The maximum utility is $2.27*10^{26}$. As a result, we may conclude that optimal ABS ratio for this system is 0.42 (42%), which solves the first issue of applying eICIC.

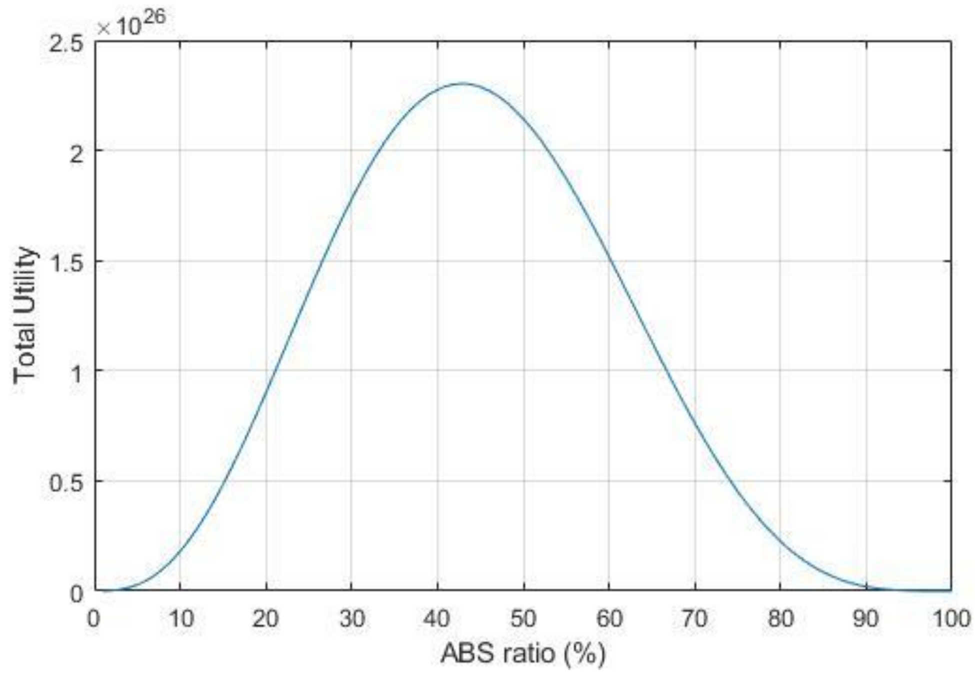

Figure 3-1 Total Utility changes with ABS ratio

The most important parameters, ABS ratio for eICIC, have been decided by now. Nevertheless, we also need to evaluate the effects of eICIC for this HetNets system and apply it in current system. Instead of concerning total capacity gain for all small cell UEs, we are more interested in the effects of eICIC in different tier of UEs. It is because that each UE is unique due to its location and receiving interference, thus eICIC may not affect all UEs evenly. In order to have a detailed understanding, average UE capacity for different tiers may be more useful, and figure 3.4 shows it for first 5%, 10%, 25% and 50% small cell UEs allocated in ABS. This figure implies (1) gain is dramatically large for first 5% UEs, which is 64.3% (2) gain starts to decline as more UEs are calculated, 10% 25% and so on.

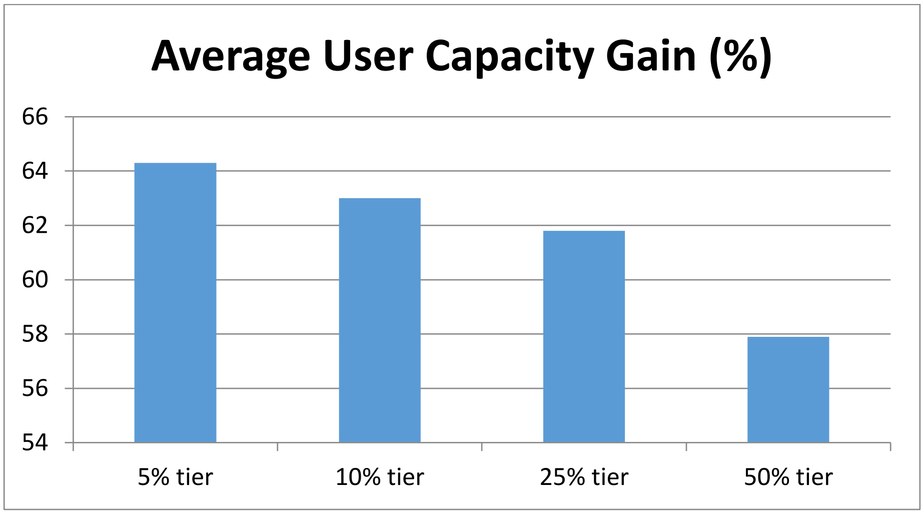

Figure 3-2 Average User Capacity Gain for different Tier of Users

This discrete figure helps to obtain exact capacity gain for each tier. In order to have a clearer observation of eICIC effect tendency figure, we draw another continuous plot of average throughput gain changes with percent of ABS UEs are calculated in Figure 3.5. Fig indicates that capacity gain gradually drops as more ABS UEs are taken into consideration. This decline rate also increases accordingly and sharply increases when last 20% ABS UEs are calculated.

Since UEs are aligned in descending order of its distance from small cell, it can be concluded from these two figures that: Cell edge users has a better benefit gain when applying eICIC, and the effect of eICIC will gradually fades as UEs are nearer to small cell (As a matter of fact, it can even be ignored by center users). This also follows our prediction in theory part.

Furthermore, this phenomenon also explains why total utility drops after *r* reaching the balance point.

We have discussed in theory part that, 2-player NBS equation requires both players to reach a balance point instead of inclined to any of them. If either part is over-extended, the other part will be forced to drop due to limited resources, and total utility will suffer a heavy loss. Considering the equation of P2 (3.3), r and total protected UEs is in numerator and denominator separately. If ABS ratio increases, total number of UEs allocating in protected subframes must increase accordingly to maintain balance. Similar situation also happens in P1 (3.2). By considering these conclusions, relationship between ABS ratio and utility shown in Figure3.3 may be explained. As ABS ratio increases, cell edge users will be allocated in protected subframes first, and utility will keep increasing because these UEs have a large eICIC benefit gain; the effect of eICIC will keep fading as UEs with lower gain are forced to enter protected subframes; when ABS ratio reaches 0.42, the benefit of eICIC in new UE is so small that it will start to hinder total utility. As ABS ratio moves from 0.42 to 1, total utility keeps decreasing as more ‘innocent’ UEs are forced to enter protected subframes. As a result, 0.42 is the balance point which maxes total utility.

3.3.4 eICIC Summary

For simulation 1, we have managed to formulate our schemes to obtain the important eICIC parameters - ABS ratio and the corresponding UE partitioning ratio can be decided by algorithm 1, so that eICIC can be applied to current HetNets system. Furthermore, simulation result shows a promising user capacity gain by mitigating cross-tier interference, especially in edge users. Meanwhile, several interesting phenomena and theories are also discussed, which may help us to set up the next simulation.

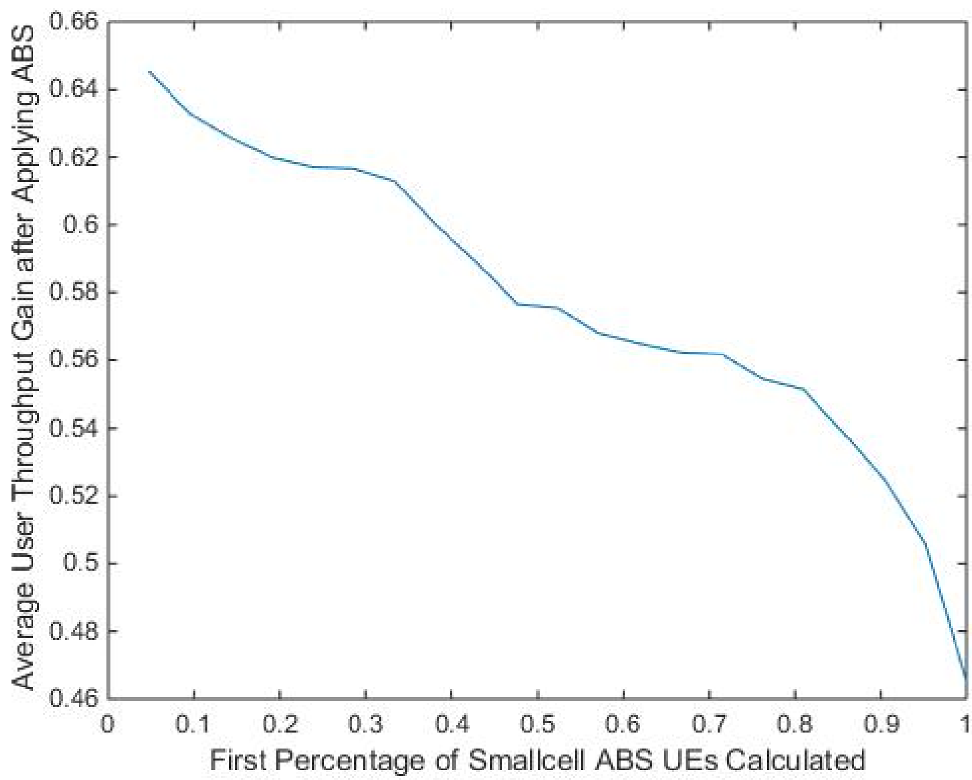

Figure 3-3 Continuous Average User Capacity gain changes

3.3.5 Simulation of FeICIC: Case 2

For this simulation, a two-tier HetNets system is built up with one central macrocell of circle coverage for simplicity, and the radius is 500 meters. 4 small cells are allocated in four quarters of the circle with 300 meters away from macrocell. 500 UEs are randomly distributed within the macrocell coverage circle. Initial CRE =11 dB is set to offload UEs.

Since basic parameters configuration is the same to case 1 simulation, only changing parameters are shown in table 3.2.

Table 3-2 Important simulation parameters for Simulation 2

Parameters	**Value**
CRE	*11 dB*
rp-ABS types	*2*
Total ABS ratio (sum of both types)	*2%-80%*

3.3.6 Simulation result and analysis

After simulation setup, we apply proposed 6-layer NBS Algorithm to find optimal combination of 0.67-ABS ratio, a_1 and 0.33-ABS ratio, a_2, which maximize the total utility. Figure. 6 shows how total utility changes with $a_1$and a_2. According to the figure, total utility will reach maximum 17.89×10^{10} when a_1= 0.41 and a_2 = 0.36.

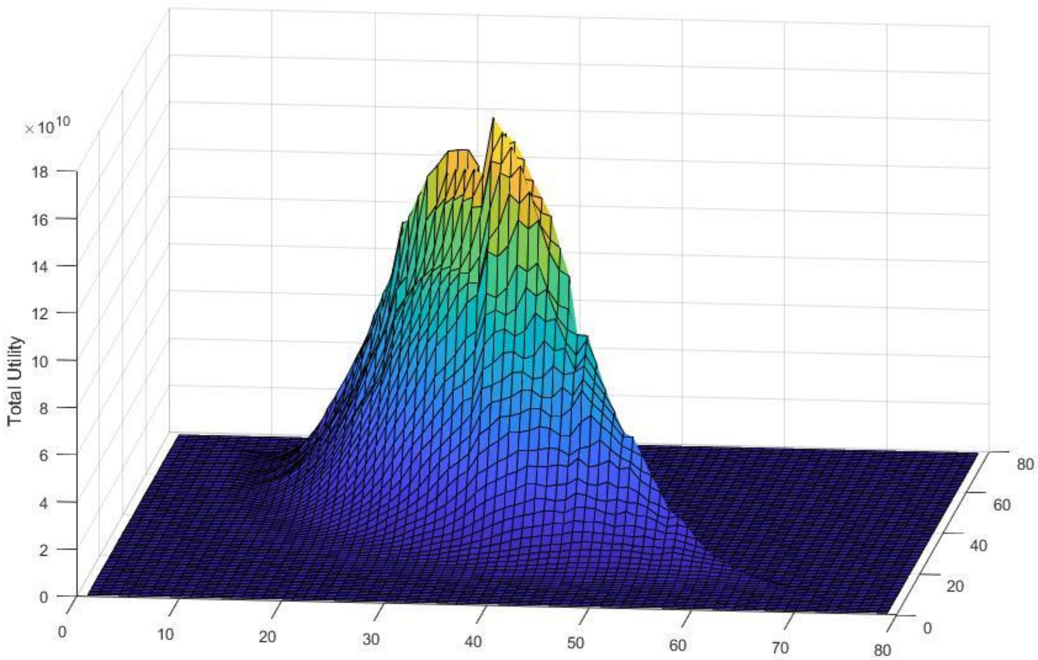

Figure 3-4 NBS Total Utility changes against a_1 *and* a_2

3.3.6.1 Macrocell UE analysis

As mentioned in introduction section, the benefit of our proposed layer-FeICIC over eICIC is to increase the efficiency of macrocell by power control instead of muting it during ABS. Unlike eICIC, UEs in macro cell will also compete to get better subframes, and therefore take part in the NBS calculation. As a result, the benefit of our proposed FeICIC algorithm should reflect in two aspects: the *system capacity* for group and *QoS* for each single macro UE.

Figure. 3.7 shows macro UE data rate after applying our proposed FeICIC algorithm. Figure. 3.8 shows macro UE data rate after applying eICIC algorithm. We align macro UEs in descending order according to their distance from macrocell for these two figures. Therefore, the first part of the figure are center UEs which are nearest to cell and are allocated in 0.33 rp-ABS. Although the transmission power during these subframes are reduced up to 66.7%, UE in this layer still has a data rate gain. This is because that these UEs are near to macro cell and far away from small cell, so that they are not vulnerable to cross-tier interference. By travelling a small amount of distance, their SINR will not suffer a dramatic drop yet, and the additional subframes (although power is low) will boost the data rates. The UEs in second part and third part will enjoy a moderate power and a better data rates gain. It can be observed that, after applying FeICIC, almost no UE's capacity is below 100 kb/s instead 65.5% UEs in third part are below 100 kb/s when applying eICIC, and the third part of the macro UEs benefit from FeICIC most. Furthermore, the UE data rate for part 2 and 3 are maintained in a moderate level, there is no sudden capacity drop for each single UE. This is because of the structure of step layer. Through gradually reducing power ratio as stepped structure, SINR can be efficiently controlled so that no single UE will suffer severe QoS drop compared to eICIC which is the second benefit of layer-FeICIC shown in macro cell.

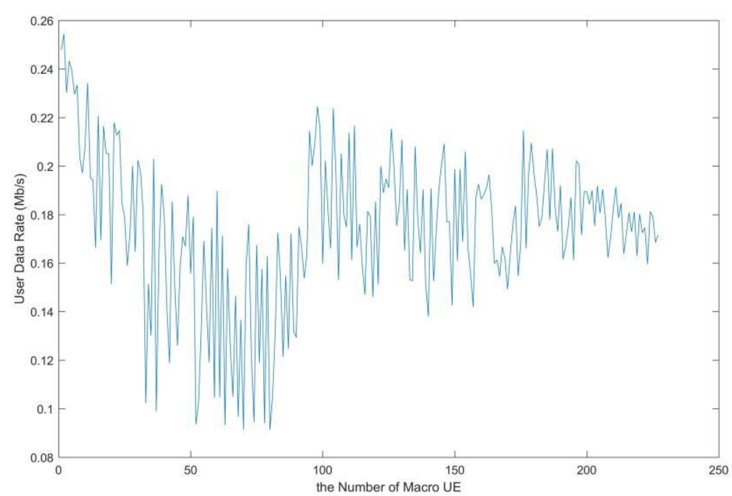

Figure 3-5 Macrocell UEs' capacity with proposed FeICIC

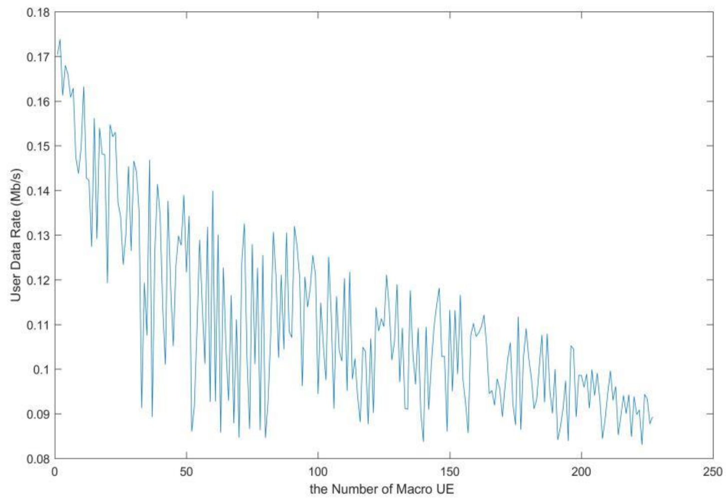

Figure 3-6 Macrocell UEs' capacity with eICIC

3.3.6.2 Small cell UE analysis

After considering the benefit of layer-FeICIC on macrocell UEs, we should evaluate how our proposed algorithm affects small cell UEs. Similar to ABS, the effect of rp-ABS on small cell UEs is to mitigate cross-tier interference. The effect is more obvious on cell edge users because their RSS from small cell is small and are more vulnerable to cross-tier and intra-tier interference. Therefore, we will focus on how FeICIC works on 10% tier UEs. Besides that, we have stated that CRE bias may greatly affect the performance of FeICIC because it controls the number of offloaded UEs. We have set CRE to be fixed in Simulation 1 for simplicity. For now, we will analyze the effect of CRE and find the optimal value for the system.

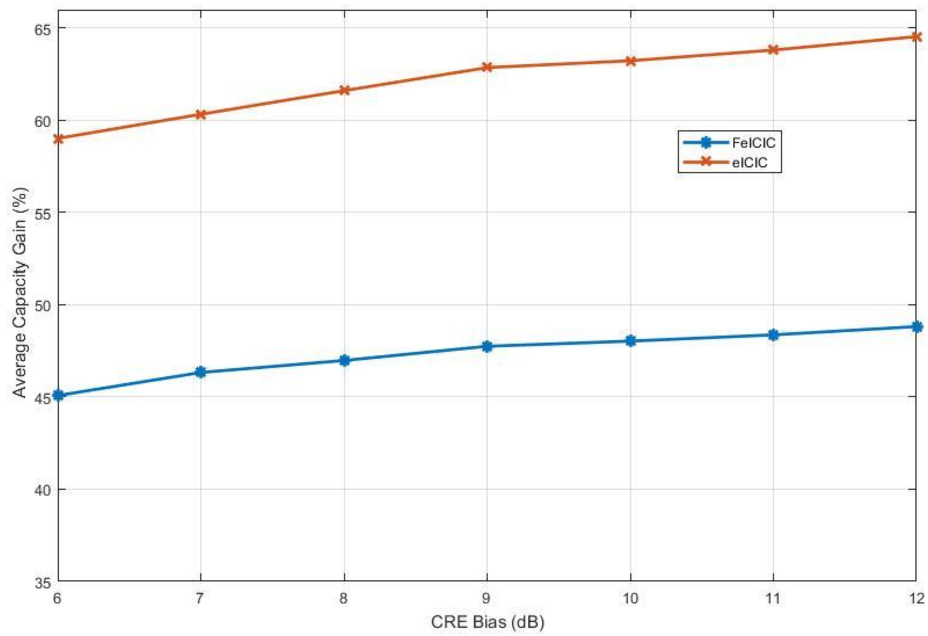

Figure 3-7 Average capacity gain for 10% tier cell edge users

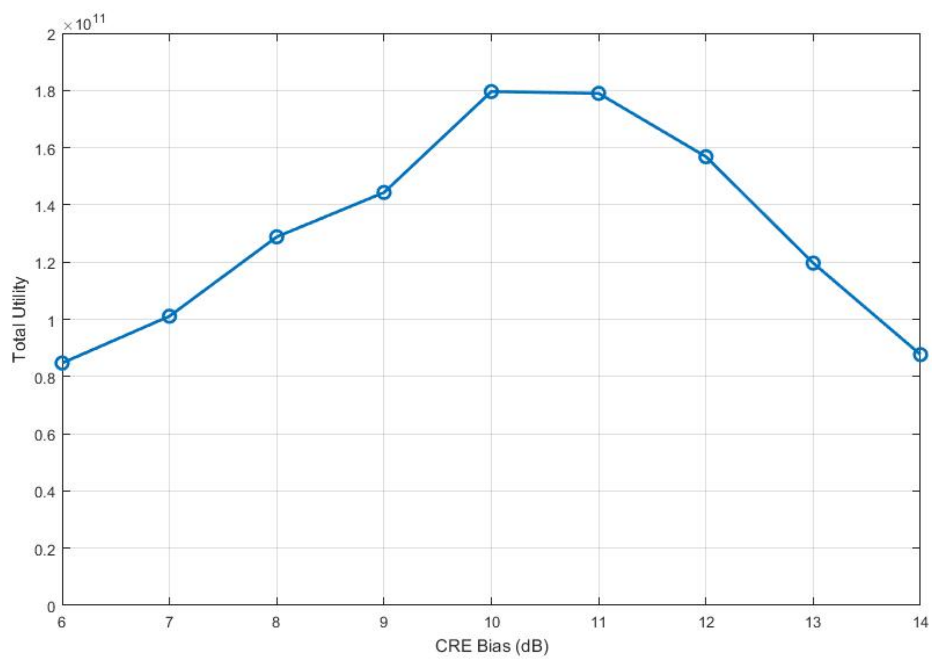

Figure 3-8 Total Utility changes with CRE bias

Figure 3.9 shows the average capacity gain for 10% tier cell edge users after applying our proposed eICIC (a=0.42) and FeICIC under different value of CRE. Firstly, we can observe that the capacity gain for eICIC is larger than FeICIC, this is because that the macro cell is not 'mute' during rp-ABS, and there is still partial of cross-tier interference. However, the effect of eICIC is also limited because intra-tier interference still exists and the advantage of FeICIC is compensated from macro cell UEs. Secondly, figure 3.9 implies that the increase of CRE bias will bring capacity gain for edge UEs. We have known that CRE is normally used to offload UEs. With the increase of CRE, more and more macro UEs will be forced to bound to small cell. These UEs will suffer severe cross-tier interference because CRE has changed the user association condition. Therefore, the capacity gain will also increase with CRE bias which explains the Figure3.9. However, this does not mean that CRE should be set the higher the better. If CRE is too high, not only the cell edge user of macro cell will be offloaded, the middle range UEs with low RSS from small cell will also be affected. Even if the cross-tier interference can be mitigated by rp-ABS, these UEs will still suffer from high intra-tier interference because their small cell RSS is too low. Considering from utility aspects, macro cell edge UEs will serve better for total utility if they are offloaded before CRE reaches balance point. When CRE increases and crosses the balance point, more and more innocent UEs will be offloaded to small cell and total utility will start to drop. Figure3.10 shows how total utility changes with CRE, which follows our analysis. Therefore, CRE=10dB may be the optimal value for our simulated scenario.

Moreover, we have known that UEs in edge user groups still has different degree of vulnerability to cross-tier interference due to increase of CRE. Therefore, if we can quantilize the users into more groups and introduce more layers, the total utility of the system can be further enhanced. The actual layer number can be decided according to operators and device situation as more layers will inevitably increase the computing and coordination complexity and hence the operating expenditure.

3.3.7 Comparison of FeICIC and eICIC

After separately discussing how FeICIC affects macro UEs and small UEs. We can conclude that the gain of FeICIC system capacity compared to eICIC is the gain from Macro cell minus lost from Small cell. If the optimal total utility is reached, there should be a system capacity gain, because FeICIC requires not only small cell UEs but also macro cell ones to take part in NBS calculation (a cooperation among whole system unlike eICIC).

In order to evaluate our proposed algorithm, we draw four CDFs for user data rate with different parameters on Figure 3.11. Compared to conventional eICIC configure, which *ABS ratio*= 0.5, *reduced power ratio* = 0, proposed FeICIC has a dramatic advantage after 0.1 Mb/s. This leading situation continues even in high data rate, which is over 0.2 Mb/s. Meanwhile, eICIC only shows better performance before 0.1 Mb/s. We thus calculate the total throughput for both scheme to make comparison, and result shows that FeICIC has a total user capacity gain as 31.4% compared to eICIC. The advantage of eICIC in lower and middle range is because of the benefit of total 'muting' of macro cell. However, the advantage may not be dramatic because intra-tier interference still exists. The advantage of FeICIC in higher range is because of the improved macro cell UE throughput. Due to high transmission power of macro cell, the extra subframes will lead to higher user capacity compared to small one. And our proposed layer structure provides further coordination of these extra subframes besides ABS in terms of power layer. Therefore, our scheme will mitigate the contradict between cross-tier interference and reduced power subframes, and hence achieve higher total throughput compared to eICIC. However, the achieved utility of our proposed FeICIC may be much lower compared to eICIC. The product of our total User utility is 17.89×10^{10}, while eICIC can reach up to 2.78×10^{26} . The reason is mainly because our FeICIC scheme has introduced opportunity cost system so that the actual utility gain for each user will drop dramatically compared to no-cost partial fairness system of eICIC. And then we compared the fixed power ratio FeICIC with our proposed dynamic step layer FeICIC. By even two types of rp-ABS, the parameter of total *ABS ratio* and *average reduce power* can be calculated as 0.77 and 0.52. We plot the CDF with new parameters and shows it on Figure 3.11. It is obvious that overall performance of proposed algorithm is better, although there is some fluctuation in lower and higher range. Besides that, we have observed that there are still three local maximums for total utility in Figure 3.6, which are caused by the local max utility of different small cells. In order to test whether our proposed parameter is the optimal, we have also plotted the CDF for these three local maximums in Figure 3.12. Our proposed parameter shows an obvious advantage although the high level capacity may have some fluctuation, which is caused by macro cell UEs.

As a result, FeICIC has a total user capacity gain as 31.4% compared to eICIC.

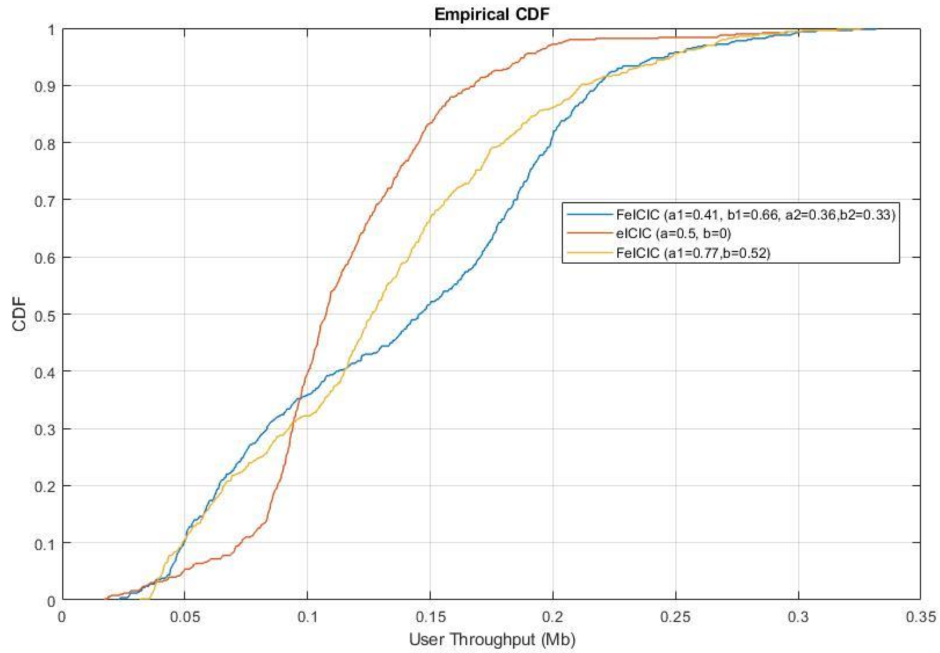

Figure 3-9 CDF for different parameters of eICIC and FeICIC

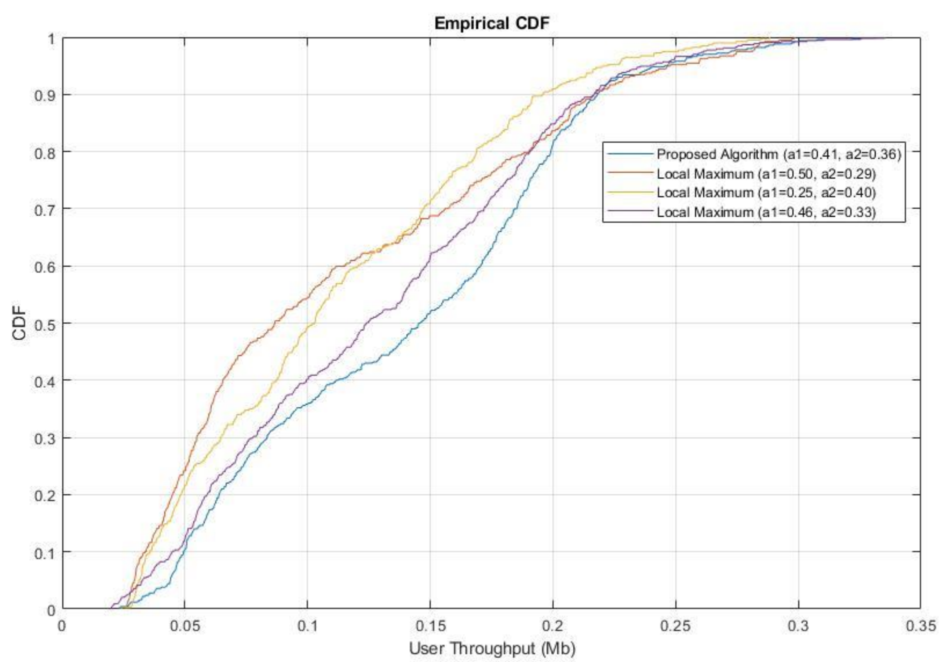

Figure 3-10 CDF for different local maximum

3.4 Conclusion

In Chapter 3, we have managed to solve cross-tier interference issue of HetNets with Almost Blank Subframes. Through muting macro cell in specific ABS, small cell UEs will benefit from it without cross-tier interference. This chapter firstly apply Nash Bargain Solution with proportional fairness to determine the optimal ABS ratio and UE allocation. Which UE are more vulnerable and how ABS affect small cell UEs are also discussed. With the information from ABS, we propose the Power-Layer Based NBS algorithm to realize reducing power ABS. During Re-ABS, macro cell power is no longer fully muted, we implement the cost of NBS according to power layer and introduce stepped power reduction, so that both the small cell and macro cell UEs may enjoy a system balance. The optimal Re-ABS ratio and UE allocation for different layer subframe is obtained and evaluated in the end.

Chapter 4. HetNets Offloading Issues Management with Machine Learning Algorithms

4.1 Introduction

So far, we have managed to solve the interference issue of HetNets by applying ABS and rp-ABS. In this chapter, we will focus on solving the second issue – offloading. This includes two aspects: 1) offloading macro cell UEs to mitigate load unbalancing due to topology of HetNets and 2) offloading large amount of small cell backhaul data from backbone network. As discussed in literature review chapter, machine learning algorithms may help to train a solution to solve both offloading problems. Therefore, we have discussed both scenarios in this chapter and propose corresponding machine learning solutions.

In first part, we proposal a modified K-means Clustering Algorithm (KCA) to offload UEs from macro cell, which we call user-based K-means clustering algorithm (UBKCA). This proposed algorithm uses the HetNets' background information and applies the eICIC technique to find the optimal CRE bias for given scenario and offloading objective. In particular, the central user group set is established to reduce computation complexity and CRE bias is introduced to enhance the performance of the algorithm in the offloading factor. Next, to realise classification and prediction for new elements entering HetNets, two methods have been applied for comparison. The first method obtains the decision boundary for current clusters through linear classification, while the second method applies the K-NN classification algorithm to the obtained clusters. The protocol for future self-optimisation is also established so that the system can realise self-organisation in UE partition and adapt to the change in traffic map automatically.

For second part, we have continue to assess KCA's capability in partitioning elements with similar parameter patterns into clusters. Since users may have different content preference patterns, we propose a proactive caching algorithm based on KCA to offload backhaul traffic into small cell network with local cache server. Through clustering users with same data preference and installing respective small cells with cache, the second offloading issue will be mitigated. Then, the K-NN algorithm is applied to predict the new UEs' group and test their backhaul offloading situation. Finally, a recommendation system capable to predict missing popularity score is established.

4.2 Methodology

4.2.1 K-Means Clustering Algorithm (KCA)

As discussed in literature review chapter, K-means Clustering Algorithm has already been widely used to solve the partition problem in various fields [129]. It is considered capable of clustering Big Data because of its fast speed, automation, and high adaption. Therefore, this paper will first apply K-means clustering to decide user association in HetNets, which will solve the first issue. In general, KCA can be formulated as a mathematical computation problem P:

$$Minimize\ P(M, C) = \sum_{i=1}^{k} \sum_{j=1}^{n} M_{i,j}\, d(R_j, C_i)$$

$$Subject\ to\ \sum_{j=1}^{k} M_{i,j} = 1,\ 1 \le j \le n$$

$$M_{i,j} \in \{0,1\},\ 1 \leq i \leq k \tag{4.1}$$

We need to divide a dataset $R = \{R_1, R_2, \ldots R_n\}$ into k clusters, and $C = \{C_1, C_2, \ldots C_k\}$ is the set of all cluster centres. i means iteration of all elements within data set, j means iteration of all clusters. M is an $n \times k$ matrix that presents all elements' decision, and $M_{i,j}$ is either 0 or 1. KCA will then be applied so that the partition will minimise (3.1). For our scenario specifically, all system UEs can be mapped as dataset R, and number of network tiers can be mapped as number of clusters K. Therefore, offloading problems can be transferred to clustering problems so that each UE can find its optimal network tier – macro, pico, femto and so on. For implementing KCA, we need to analyse the data in the following steps:

- Find a suitable partition number by analysing data, which means the data will be divided into k groups.
- Randomly select k elements and assume them to be the centre of each group.
- Allocate all remaining n-k elements to their nearest centre to form clusters, and then calculate the new centre for each cluster.
- Repeat the steps till the system converges.

To reduce the computation complexity of the algorithm, we use the Euclidean distance method to calculate the total distance between the elements and centres in Equation (1). The equation is as follows, where $p_1, p_2 \ldots p_n$ represent all elements within the data set and q_1, $q_2, \ldots q_n$ represents the corresponding clustering centres [124].

$$d(p,q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \cdots + (p_n - q_n)^2} \tag{4.2}$$

4.2.2 K-Nearest Neighbouring (KNN)

K-NN was first developed in 1968 by Cover and Hart [131]. The basic concept of the model is as follows: Transfer the collected data into a multi-parameter vector, with parameters for each element recorded with the score according to their weight. The vector can be shown as $D = (V_1, V_2, V_3, \ldots, V_N)$. Each V is the vector of parameters for the respective element. We store this dataset as a training dataset, and then, compare the test data with it to find the K-nearest (according to distance) or most similar (according to similarity) elements from the training dataset. The similarity calculation formula is shown below [132]:

$$Sim(e_i, e_j) = \frac{\sum_{n=1}^{M} V_{in} \times V_{jn}}{\sqrt{\sum_{n=1}^{M} V_{in}^2}\sqrt{\sum_{n=1}^{M} V_{jn}^2}} \tag{4.3}$$

Where e_i is the test data, e_j is the comparison data obtained from the stored training dataset, M is the total number of parameter types, and n represents the current parameter type. K in K-NN indicates the number of similar elements we want for voting. After obtaining K elements, we will test their weight in voting. A typical method is as follows:

$$W(e_j, C_i) = \begin{cases} 1, & \sum Sim(e_i, e_j)\, a_t - Threhold > 0 \\ 0, & else \end{cases} \tag{4.4}$$

Where Ci represents the ith cluster, which is being tested. a_t is 1 if e_j belongs to Ci; otherwise, it is 0. T is the threshold to filter out irrelevant elements if the total similarity is not high enough so that both accuracy and computing complexity can be moderately maintained. By calculating the votes of all K nearest elements, we can classify the test data into the corresponding cluster.

The method above is one way how to decide the 'distance' among elements, which is to calculate the similarity. This method is useful for elements with many parameters or in specific score system, which will be discussed in later section. For elements with few parameters (2 is a typical situation), Euclidean Distance Method is also a straight and efficient way. It is because the similarity is not reliable without enough parameter comparisons.

4.2.3 Combination of KCA and KNN

KCA is an *unsupervised* learning algorithm specifically used to solve the clustering problem. By analysing the characteristics and comparing the dissimilarity of a group of data, K-means can divide them into k groups and iteratively make new decisions as the group grows without further 'supervision' [133]. In other words, this unsupervised algorithm will efficiently reduce the computation complexity because it can adapt to the rapidly-changing user traffic map through self-organising, and thus, is suitable for solving the first issue.

However, the K-means algorithm will only perform clustering according to the data's characteristic but also faithfully reflect the real-time situation. Most UEs will still be allocated to the macrocell group due to its high transmission power advantage. As a result, we need to further modify the algorithm to fit our HetNet scenario, and CRE may be the key to solve this issue. CRE is a technique of eICIC, and was first introduced in 3GPP Release 10. It affects

UEs' association decision by adding a virtual bias to the power received from the small cell. Therefore, the situation of load unbalancing can be mitigated by applying CRE.

After clustering the existing UEs through K-means algorithm and realising UE offloading in HetNets, the next step is the prediction of the new UEs' association when they enter the system. Classification is a category of *supervised* ML algorithm, which is a popular research field and has been widely applied in pattern recognition, statistics, medical science, etc. [134]. Similar to traditional supervised learning algorithms, classification requires a list of target result variables combined with the predefined parameters. The ML model analyses this training dataset along with their parameters or predictor variables and finds the optimal pattern. If the training dataset is provided in clustering type, such as categorised files, this model can classify new elements into this clustering according to its parameters, so the prediction can be realised. K-NN is a popular classification algorithm, which can also be considered as instance-based learning [135]. Before classification, the model should have a set of training data as reference. During the classification phase, the newly entered element is compared with the stored reference training data and classified into the cluster with the most similar parameter pattern.

After understanding the functions of KCA and K-NN, we can combine these two ML algorithms to implement a SON system for HetNets UE partition. First, we can apply KCA to the existing UEs and obtain the UE clustering pattern and store it. Second, we can apply K-NN and use this stored UE clustering pattern as a reference training set. As a result, any unclassified UE entering this HetNets system will be rapidly assigned to its suitable tier network without further human intervention, thus reducing the computation complexity.

However, K-NN still has some issues before application to the HetNets system. First, K-NN normally considers all neighbouring elements to have the same importance when countering the number of votes from the stored training dataset [136]. Therefore, neighbouring elements from different clusters may contribute the same vote weight. This is clearly not the case for HetNets, especially for classification between macrocell tier and small-cell tier, due to the unbalanced transmission power. Second, once the new element is assigned to the cluster according to its parameter pattern, there is no further definition of this element. As a result, the effect of this new element on the respective cluster pattern can be neglected. Nevertheless, if the number of new elements is sufficient, the effects may be accumulated to change the stored cluster pattern [134]. This issue is even more severe for our scenario because we have discussed that the network is designed to adapt to the rapidly changing traffic map. As a result, these two issues should be addressed if we want to apply it for UE classification in HetNets.

4.2.3 *Distance Normalisation*

In KCA, we map the similarity between two elements into the distance between them. The less the distance, the more similar are the two elements. Therefore, the centroid of the cluster is more like an example for elements within. As mentioned above, we can use the Euclidean distance method to calculate the distance between two elements when making decisions. However, we cannot make sure that all parameters are comparable with each other. Certain parameters may be much larger than others and play a dominant role when calculating the distance, such as yearly revenue compared to age, while some parameters' value is more like a symbol and is difficult to link to other statistical values, such as years (e.g. 2017). As a result, we have summarised several methods to normalise the parameter values and the formula is shown as below [137].

Min-Max Normalisation:

$$e' = \frac{e - \min(E)}{\max(E) - \min(E)} \tag{4.5}$$

Z-score Standardisation:

$$e' = \frac{e - mean(E)}{SD(E)} \tag{4.6}$$

Quantisation Normalisation:

$$e' = \begin{cases} 1, & if\ e > Threhold \\ 0, & else \end{cases} \tag{4.7}$$

Log Normalisation (dBm):

$$e'(dBm) = 10 \log_{10}(1000 \cdot e(Watt)) \tag{4.8}$$

From (4.5) to (4.8), e means original element, e' means normalised element, E means the data set of all elements. Each normalisation method has its advantage. Min-max normalisation will normally transfer the original value into a value between 0 and 1 only, where $\min(E)$ means minimal element within data set and $\max(E)$ means maximal elements within data set. The normalised value of Z-score normally lies between $-3 < z < 3$, which has a wider distribution. And $SD(E)$ means Standard Deviation of data set. Quantisation normalisation will map all values into either 0 or 1, which is more suitable for a logical problem. Log normalisation (dBm) will transfer data from Watt into dBm, which is often used in plotting the figure in wireless communication. $e(Watt)$ means element with unit of *Watt* and $e'(dBm)$ means unit of *dBm*. Although the unit is mapped for benefit of plotting, we will still use e(Watt) to calculate Euclidean distance.

4.2.4 User-Based K-Means Clustering Algorithm (UBKCA)

For our scenario, we define elements in the system to be UEs in HetNets. Each UE is assigned with two parameters so that they can be analysed by KCA. Since UE prefers to choose a cell with better signal reception, we define these two parameters to be receiving power from a macro cell (*Rm*) and small cell (*Rs*), respectively. In order to have clearer simulation result, we have recorded the parameters in dBm formation.

One issue of KCA is that it starts with random centres, and thus, may require several iterations before the system converges, which obviously increases the computation time. Besides, if the initial centres are chosen to be too close, the final result can only be a local solution instead of a global one. Therefore, we decide to modify the KCA with essential background knowledge to avoid redundant computing. For central users of a macrocell, its *Rm* is much larger than *Rs* because of their short distance to the BS. Therefore, macrocell central users will not be offloaded to small cells when HetNets are established, and vice versa. By introducing this property in our KCA algorithm, we can significantly reduce the computation time. We define a special group data element with similar parameters as ‘central’ users. Whenever an element in the group is distributed to a cluster, the remaining group elements are automatically assigned to the same one. All group members have proven to be ‘loyal’ because of their central position, and no more computation is needed even for the next several time intervals. For example, since the receiving power is distance-based, we have defined top 20% users with highest receiving power from a macrocell to be its central user group. This method has a clear advantage: not only can the computation time be reduced but the local maximum can also be avoided.

After analysing the property of HetNets, we modify KCA by introducing the central user group to reduce the computation complexity, which can solve the first issue we mentioned in the Introduction part. However, the result of KCA might not be practical due to the unbalanced transmission power for different tiers. To solve this second issue, we add another modification to our proposed KCA, which is to apply CRE during the KCA process.

As described in first part, CRE bias can be added to UEs’ small-cell receiving power to help UEs offload from a macrocell tier to a small-cell tier. However, not all UEs will benefit from CRE affection, especially for the central users. Their *Rm* is so large compared to *Rs* that they will still choose to stay in the macrocell even if the CRE bias is added. As a result, adding CRE bias to all UEs is not realistic because it increases computation time and degrades efficiency. Therefore, we choose to apply CRE to edge users only instead of central users to increase computing efficiency. We set up an edge user factor *alpha* (α) to decide which UE should be considered as edge users.

So far, we have established the model and applied KCA to classify UEs into two tier groups. Furthermore, we have made two modifications to enhance KCA performance: central user concept and CRE bias. Since both modifications are applied to user data, we call our proposed algorithm as UBKCA, which can be formulated as follows:

Algorithm 4.1: UBKCA

Input:

- Training Data set *R* with total element number N_1, each element within has various parameters *Rs, Rm ….*
- Different cell tier center user subsets (C_m, $C_{s,\ \ldots} \subseteq R$)
- Initial Edge user factor α
- Initial CRE bias β
- Number of clusters, *K*
- Set system error $E = 0$, and corresponding temporary variable $E' = 0$
- Set iteration step variable i =1, j=1

Initialization and Iterations

1. **for** $\alpha = 0.05$ to 0.25 **do**
2.   **for** $\beta = 1$ to 10 **do**
3.   **while** $E > 0$ **do**
4.   Randomly select *K* elements as initial cluster centroids (c_1, c_2, … c_k)
5.   **for** $i = 1$ to N_1 **do** (loop for all elements within training set)
6. I_i = index of cluster whose centroid has the minimum Euclidean distance to current element, R_i
7.   **if** $R_i \in C_m$ **then**
8.   Index of elements in $C_m = I_i$
9.   **end if**
10.   **if** $R_i \in C_s$ **then**
11.   Index of elements in $C_s = I_i$
12.   **end if**
13.   **end for**
14.   **for** j = 1 to K **do** (loop for all clusters)
15. μ_j := mean of all elements assigned to current cluster, $G_j = \{\, Ri \mid I_i = j\}$
16.   **end for**
17.   Calculate error for current partition $E' = \sum_{j=1}^{K} \sum_{i \in G_j} ||R_{i,j} - \mu_j||^2$
18.   **if** $|E'-E|>0$ **then**
19. $E=E'$
20.   Return to step 5 with new cluster centroids as (μ_1, μ_2, …μ_k)
21.   **end if**
22.   Record offloading situation and agreement for current combination of α and β
23.   **end for**

24. **end for**

As mentioned previously, the randomness in choosing initial cluster centres may lead to a local solution instead of a global solution after applying KCA. Therefore, we need to find a method to evaluate the capability of UBKCA in classifying UEs. We apply the Rand index to measure the difference between our calculated partition and predicted optimal partition. For a dataset with n elements, if we want to choose two elements from it, there should be $\binom{n}{2}$ possible selections. After the classification, these two elements are either assigned to the same cluster or separated into two clusters (the situation may differ for various clustering methods). Suppose two partitions have been achieved with traditional user association and UBKCA; let N_1 be the number of pairs that have been allocated in the same cluster for both partitions and N_2 be that allocated in different clusters for both partitions. Then, $N_1 + N_2$ indicates all selections in which both partitions agree with each other, and (3.9) can be used to represent the degree of agreement between two partitions [8]. The agreement degree between UBKCA partition and traditional user association partition can be considered as the evaluation of the algorithm.

$$Rand(R1, R2) = \frac{N_1+N_2}{\binom{n}{2}} = \frac{N_1+N_2}{n(n-1)/2} \tag{4.9}$$

4.2.5 Self-Organization System with Supervised Algorithm

After the first step, which is to decide suitable CRE bias value offloading UEs in HetNets with UBKCA, we will establish the SON system to automatically assign new UEs entering the system with supervised ML algorithm. Besides predicting the position of new UEs, this SON system has another important objective, which is to monitor and maintain the offloading factor within a required range (40% - 60%).

According to the situation of our HetNets scenario, two potential supervised ML algorithms may be eligible for our task, linear classification and KNN. In order to find out which one is more suitable in this scenario, we have designed modified algorithms as below, and further evaluation will be applied in simulation and an analysis section.

KNN algorithm is a lazy supervised machine algorithm, therefore the pre-stored training set is essential for it. We combine UBKCA with KNN, and store original data set with corresponding label set (generated by UBKCA) as the training set for KNN.

Algorithm 4.2: UBKCA-KNN

Input:

- Training Data set R with number of N_1, each element within has *2* parameters *Rs, Rm*
- Test data set R' with number of N_2
- User factor α obtained from algorithm 1
- CRE bias β obtained from algorithm 1
- Number of considered neighbour, $K'=5$
- Set iteration step variable i =1, j=1

Initialization and Iterations

1. Predefine the value of K' according to the scenario
2. Apply algorithm 1 to obtain Cluster index set $I = \{I_1, I_2, \ldots, I_i\}$ as classify label set
3. Combine data set R with index set I as stored training set for KNN algorithm
4. **for** i = 1 to N_1 **do** (loop for all elements of training set**)**
5.   **for** j = 1 to N_2 **do** (loop for all elements of test set)
6. $D_{i,j}$ = Euclidean distance between the i_{th} test data R'_i and the j_{th} training data R_j
7.   **end for**

8. Sort $D_i = \{D_{i1}, D_{i2}, \ldots\}$ and obtain the top K' elements that have shortest value
9. V_1 = number of small cell cluster elements within the top K'
10. V_2 = number of macro cell cluster elements within the top K'
11. **if** $V_1 > V_2$
12. R'_i is classified to small cell
13. **end if**
14. **if** $V_1 < V_2$
15. R'_i is classified to macro cell
16. **end if**
17. **end for**
18. Calculate Precision of the results
19. Calculate Recall of the results
20. $F_1 = F_1$ Score of the results
21. **if** $F_1 < 0.8$
22. Return to Algorithm 1
23. **end if**
24. **if** offloading < 0.4 or offloading > 0.6
25. Return to Algorithm 1
26. **end if**

For linear classifier, we apply Perceptron Algorithm to calculate the linear decision boundary equation by combining UBKCA.

Algorithm 4.3: UBKCA - Perceptron Algorithm

Input:

- Training Data set R with number of N_1, each element within has 2 parameters Rs, Rm
- Test data set R' with number of N_2
- User factor α obtained from algorithm 1
- CRE bias β obtained from algorithm 1
- Set iteration step variable $i = 1$

Initialization and Iterations

1. Apply algorithm 1 to obtain Cluster index set $I = \{I_1, I_2, \ldots, I_i\}$ as classify label set
2. Suppose linear hypnosis function $H = w_1 * x_1 + w_2 * x_2 + b$
3. $w_0 := 0$, $w_1 := 0$, $b := 0$, $y_1 := 1$, $y_2 := -1$, $s := 0.01$
4. **for** $i = 1$ to N_1 **do** (loop for all elements within training data)
5.   **if** H(w_0, w_1, b) and I_i has same sign
6.   return to step 4
7.   **else if** $I_i = 1$ **then**
8. $w_1 := w_1 + sx_1$
9. $w_2 := w_2 + sx_2$
10. $b := b + s$
11.   **else if** $I_i = -1$ **then**
12. $w_1 := w_1 - sx_1$
13. $w_2 := w_2 - sx_2$
14. $b := b - s$
15.   **end if**
16. **return** w_0, w_1, b
17. Pr = Precision of the results
18. Re = Recall of the results
19. $F_1 = F_1$ Score of the results
20. **if** $F_1 < 0.8$
21.   Return to Algorithm 1
22. **end if**
23. **if** offloading < 0.4 or offloading > 0.6
24.   Return to Algorithm 1
25. **end if**

4.3 Simulation and Analysis

4.3.1 Simulation Setup

In this part, we will first manage to design and apply KCA to classify UEs into macro cell group and small cell group. And then we will apply UBKCA to the same UE data, and evaluate its capability with Rand Index Method. After that, we will discuss how CRE bias *βeta* and edge user factor *alpha* affect the performance of UBKCA and try to optimize the parameters of UBKCA.

For this simulation, a two-tier HetNets system is built up with one central macrocell of circle coverage for simplicity, and the radius is 500 meters. 6 small cells are allocated around the coverage circle with 300 meters away from macrocell. 200 UEs are randomly distributed within the macrocell coverage circle. The essential simulation parameters are listed as below:

Table 4-1 Simulation Parameters

Parameters	**Value**
Bandwidth	*1 MHz*
Cell layout	*Two-Tier HetNets*
User Equipment Number	*200*
Transmit power of macro cell	*40 W/46 dBm*
Transmit power of small cell	*0.25 W/24 dBm*
Noise power	*-174 dBm*
CRE	*1-10 dB*
Edge User Factor	*2.5% - 25%*
Center User Factor	*20%*
Offloading Objective	*50%*
Number of K for K-means	*2*
Number of K for KNN	*5*

4.3.2 Applying KCA to cluster UE

After the Data Set is collected from the simulation result, we apply KCA to it to partition UEs into either macro cell or small cell group. Figure 4.2 shows the result of applying KCA.

Region 1 (black dots) is the small cell region and Region 2 (Blue dots) is the macro cell region. However, it is obvious that the load is severe unbalanced in Figure 4.2, only 15% of the UEs are offloaded to small cell network, which is far below the 50% objective. It suggests that KCA can be used to realize SON of user association decision and help to reduce computing time with proper data parameters selection, but the effects are highly limited if it is directly applied to data without further background modification.

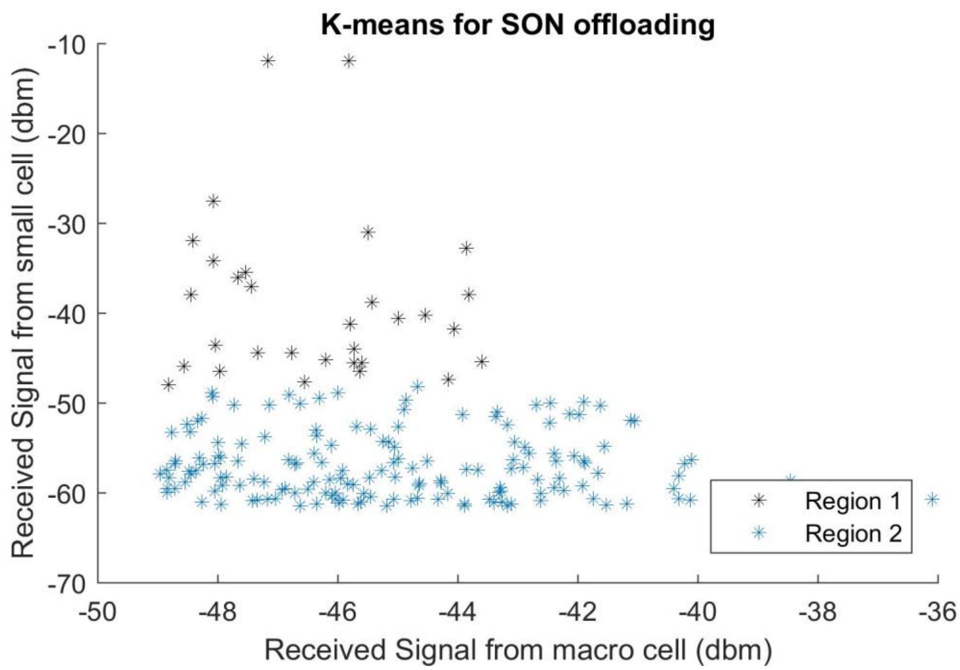

Figure 4-1 UE partition with KCA

4.3.3 Applying UBKCA to cluster UE

In order to enhance the performance of the algorithm, UBKCA is applied to the same scenario with the background knowledge of HetNets and the simulation result is shown in Figure 4.3. The first modification is to introduce two essential parameters of UBKCA – CRE Bias (β)

and Edge User Factor (α) as discussed in methodology part. Figure4.3 shows the result of applying UBKCA with *alpha = 15%* and *βeta = 6dB.* Compared to Figure 1 with KCA, the separation of two clusters are more distinct, which will lead to a better prediction of SON and faster convergence time. Besides, the percentage of offloaded UEs increases from 15% to 28%. This may be the effect of combination of *alpha* and *βeta.* Higher edge user factor means more UEs will involve in adding CRE bias, which increases the weight of small cell user cluster and will attract more UEs when computing in algorithm. Meanwhile, Higher CRE will add bias to these involving UEs so that they have more preference to be offloaded to small cell. Therefore, Figure 4.3 shows that 6dB is not enough to offload all the involving UEs to small cell cluster (part of the UEs are still assigned to macro cell group). In order to check how CRE affects offloading factor, we plot Figure 4.4 when *alpha* is maintained as 17.5%. It shows that given *alpha* is fixed, the increasing of CRE will result in the effect of *alpha* gradually consumed by the system, which makes the partition more balanced. However, we cannot conclude that we have obtained the optimal UBKCA parameters even if the offloading objective is satisfied. We also require the partition has high accuracy compared to traditional user association when CRE is applied.

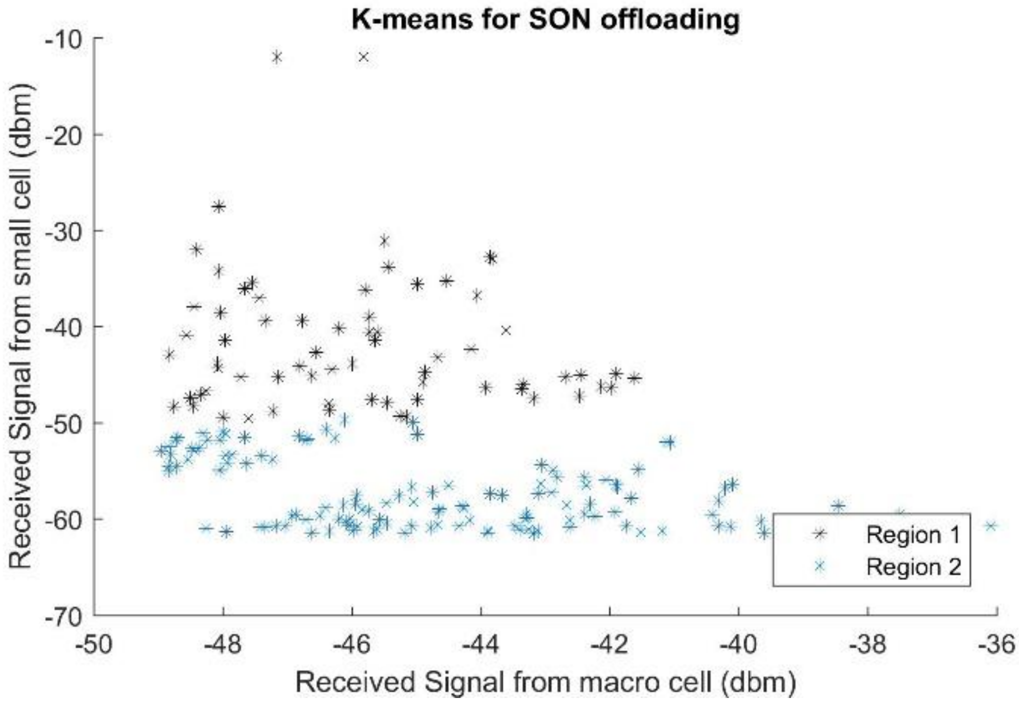

Figure 4-2 UE partition with UBKCA under α = 15% and β = 6dB

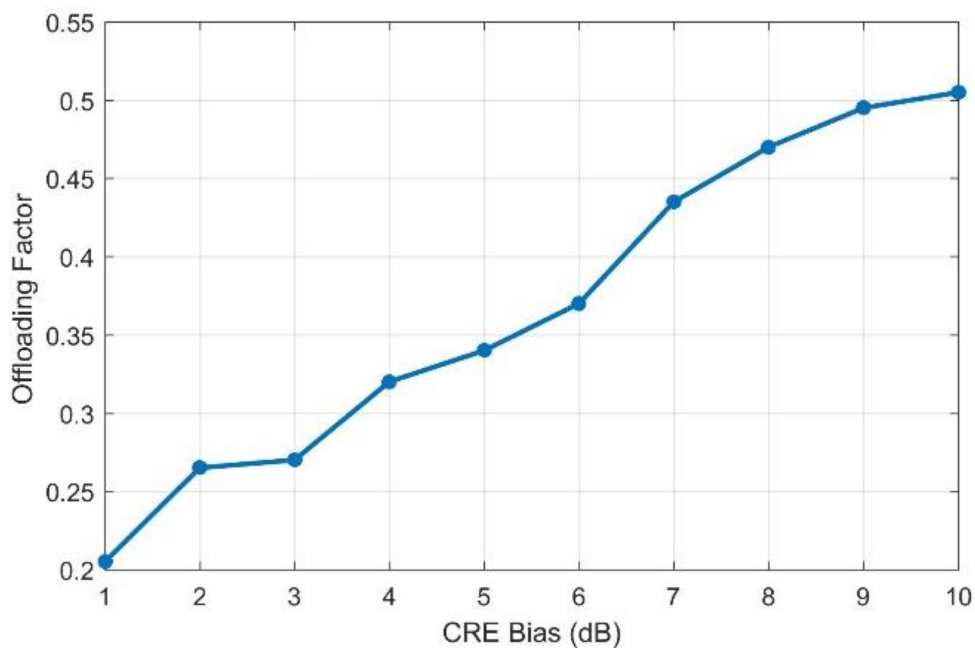

Figure 4-3 Offloading Factor changes with CRE bias under α = 17.5%

In HetNets system, higher *alpha* and *βeta* will also lead side-effect for the accuracy of UBKCA. As *alpha* increases, more UEs are forced to receive CRE bias. Eventually, the center users of macro cell will have to accept CRE bias, which not only causes the system overbalanced but also increases redundant compute time. Besides, CRE bias will amplify cross-tier interference of HetNets, which severely affects QoS of users. Therefore, we use Rand Index Method to evaluate UBKCA under specific value of *alpha* and *βeta*, so that optimal combination of parameters can be chosen. Figure 4.5 shows how agreement degree changes with *αlpha* and *βeta*, and we can observe two meaningful phenomena from it. Firstly, the average accuracy drops gradually as *alpha* increases, and the top accuracy also descended from 97.8% to 91.3% when α reaches 25%. This means that higher *alpha* will bring negative result to system partition accuracy although it will help to offload UEs to small cell network. Therefore, α should maintain low value as long as system offloading factor is satisfied. Secondly, the system requires higher *βeta* to reach top accuracy as *alpha* increases. The optimal β ascended from 1dB to 9dB when α reaches 25%. It is because that the system demands higher CRE bias to make sure the additional edge UEs can be offloaded. Once the balance is obtained, more CRE bias will not benefit the accuracy and even hinder it. As a result, within the range of *αlpha* and *βeta* which can achieve 50% offloading factor, we choose the lowest *αlpha* value 17.5%, and the corresponding *βeta* value is 8dB. The optimal clustering situation is represented in Figure 4.6.

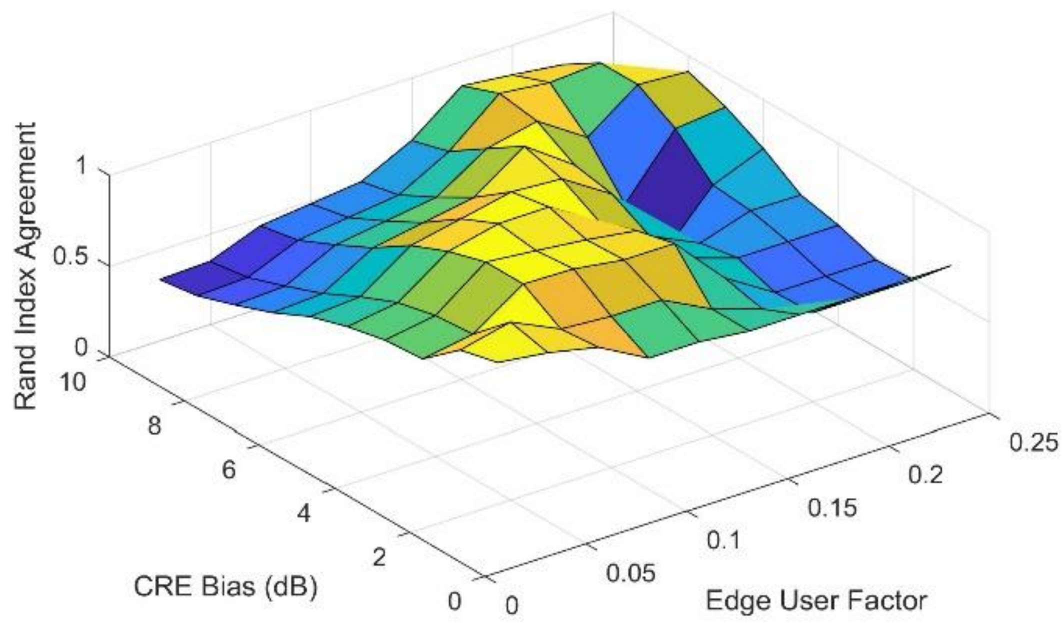

Figure 4-4 Agreement changes with a and β

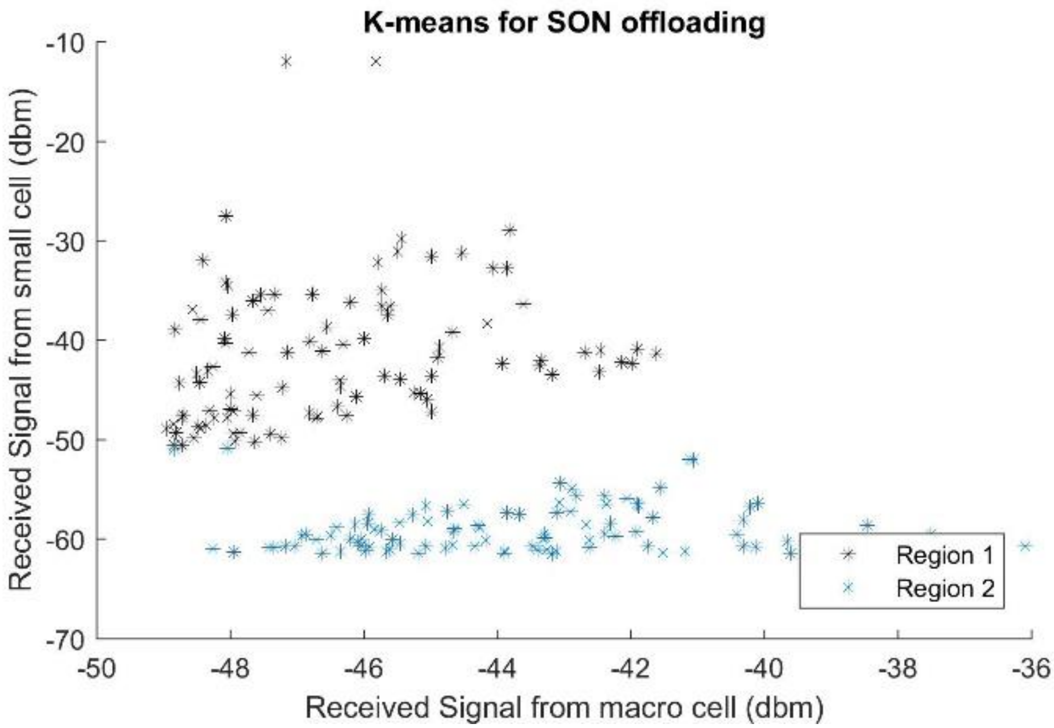

Figure 4-5 UE partition with UBKCA under optimal α = 17.5% and β = 8dB

The second modification is to establish center user group which is the 'royal' subset for macro cell mentioned in last part, so that the redundant compute can be avoided. Figure 4.7 shows the average iteration numbers changes with percentage of center users. When percentage is 0, which means the normal KCA is applied, the iteration number is around 12.8. As more users are distributed into center group subsets, the iteration number before system converges drops obviously and reaches 7.4. The total computing time will drop accordingly, which solves the second issue of applying KCA in HetNets scenario. As the complexity of network increases, collected data set will reach the scale of tens of thousands instead of just 200 UEs, UBKCA will be more capable compared to simply applying KCA.

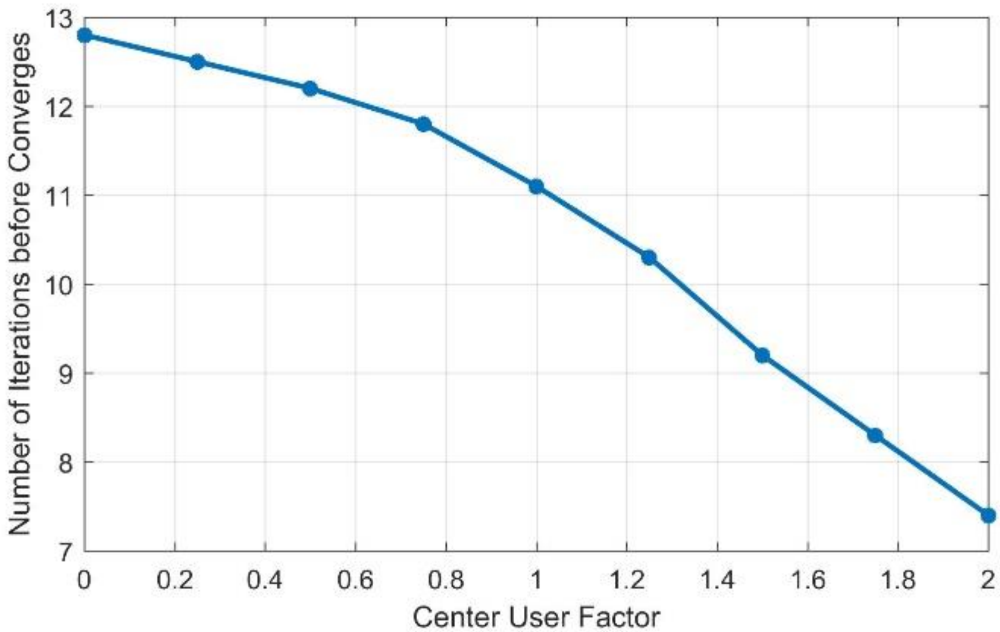

Figure 4-6 Iteration numbers changes with Centre User Factor

4.3.4 Prediction Evaluation

As mentioned in methodology part, after establishing the macro cell and small cell clustering with initial 200 UEs, we could apply supervised algorithm to predict any new UE's partition when entering current HetNets system and implement the SON system. We have introduced the linear classification and KNN method in introduction session, and stated that KNN may be more suitable in HetNets. Since we need to establish the SON system and monitor the performance to readjust the parameters if the performance is low, we may require a more stable classification method with time passing by.

In order to evaluate the performance of two classification methods, we generate the first test data with 800 UEs and made the Confusion Matrix for both methods as follows:

Table 4-2 Confusion Matrix for KNN in first test data

		True Condition		
	Total Population	Condition Positive	Condition Negative	
Predicted Condition	Predicted condition positive	True Positive = 321	False Positive = 32	Precision=0.910
	Predicted condition negative	False Negative = 63	True Negative = 384	Accuracy=0.881
		Recall =0.837		F_1 Score=0.872

Table 4-3 Confusion Matrix for linear classification in first test data

		True Condition		
	Total Population	Condition Positive	Condition Negative	
Predicted Condition	Predicted condition positive	True Positive = 313	False Positive = 42	Precision=0.879
	Predicted condition negative	False Negative = 71	True Negative = 374	Accuracy=0.859
		Recall =0.817		F_1 Score=0.847

We can analyse the performance of algorithm in two aspects. Accuracy is one contribution, but it is not reliable alone in evaluating machine learning algorithm. The main reason is that it cannot reflect the advantage if the data set is unbalanced collected. For example, if there are 90 macro cell UEs and 10 small cell UEs in the data set, the algorithm may identify all elements as macro cell. In such cases, the accuracy may reach high degree as 90%, however, this algorithm's recognition rate of small cell may be extremely low. As a result, besides accuracy, we should also calculate algorithm's ability to identify not only small UE but also macro UE, which are precision and recall. F1 score is the degree which considering both precision and recall of the algorithm, which is the more reliable metric to evaluate classification method.

The first test data set are the first 800 UEs that entering the system after K-means partition. After comparing the precision (0.910 and 0.879), recall (0.837 and 0.817) and finally F1 score (0.872 and 0.847), we cannot conclude that which one is better because there is no huge difference in the performance. However, there is another important factor we should consider when we need to establish the SON system, which is the stability.

Therefore, we should also check the adaption of these two methods with time passing by. And we generate 2 new data sets with 800 UEs each and feed into the system. The confusion matrix is as follows:

Table 4-4 Confusion Matrix for KNN in Second test data

		True Condition		
	Total Population	Condition Positive	Condition Negative	
	Predicted condition positive	True Positive = 321	False Positive = 34	Precision=0.904

Predicted Condition	Predicted condition negative	False Negative = 77	True Negative = 368	Accuracy=0.861
		Recall =0.807		F_1 Score=0.853

Table 4-5 Confusion Matrix for Linear classification in Second test data

		True Condition		
	Total Population	Condition Positive	Condition Negative	
Predicted Condition	Predicted condition positive	True Positive = 293	False Positive = 41	Precision=0.868
	Predicted condition negative	False Negative = 105	True Negative = 361	Accuracy=0.820
		Recall =0.743		F_1 Score=0.801

Table 4-6 Confusion Matrix for KNN in Third test data

		True Condition		
	Total Population	Condition Positive	Condition Negative	
	Predicted condition positive	True Positive = 315	False Positive = 41	Precision=0.885

Predicted Condition	Predicted condition negative	False Negative = 84	True Negative = 360	Accuracy=0.844
		Recall =0.789		F_1 Score=0.834

Table 4-7 Confusion Matrix for Linear classification in third test data

		True Condition		
	Total Population	Condition Positive	Condition Negative	
Predicted Condition	Predicted condition positive	True Positive = 275	False Positive = 50	Precision=0.846
	Predicted condition negative	False Negative = 124	True Negative = 351	Accuracy=0.7875
		Recall =0.696		F_1 Score=0.801

According to confusion matrix, we will be able to plot the following three figures to show how precision, recall and F1 score changes for two methods.

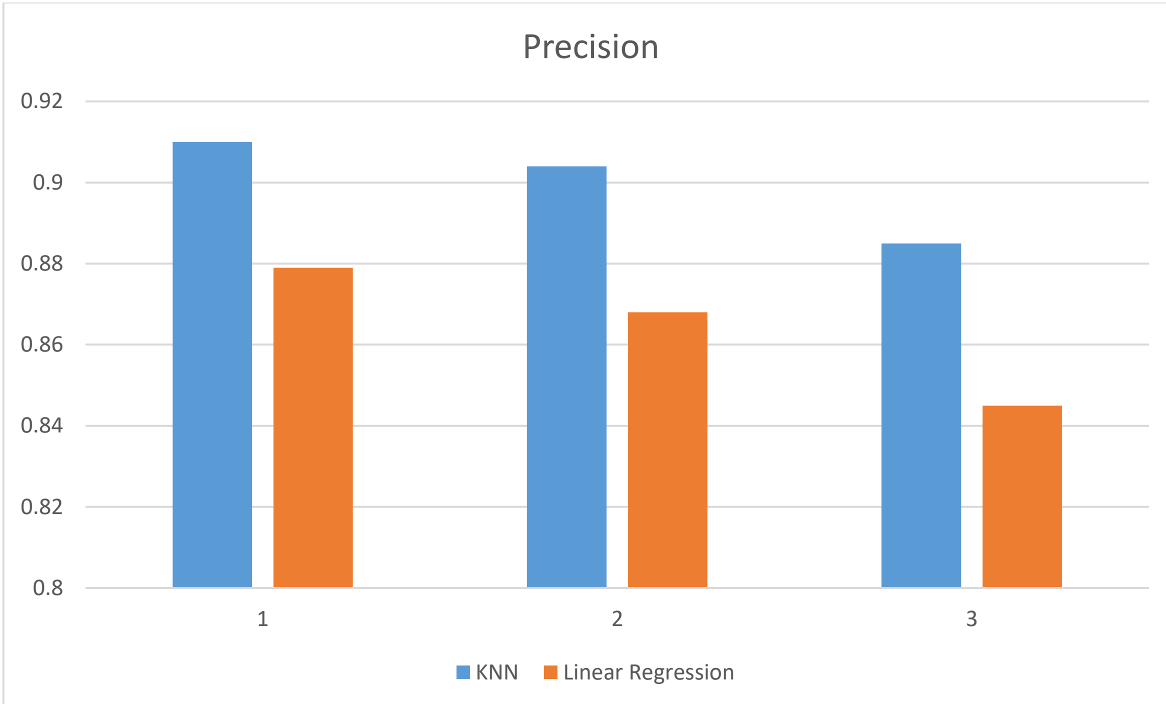

Figure 4-7 Precision changes for three successive data set

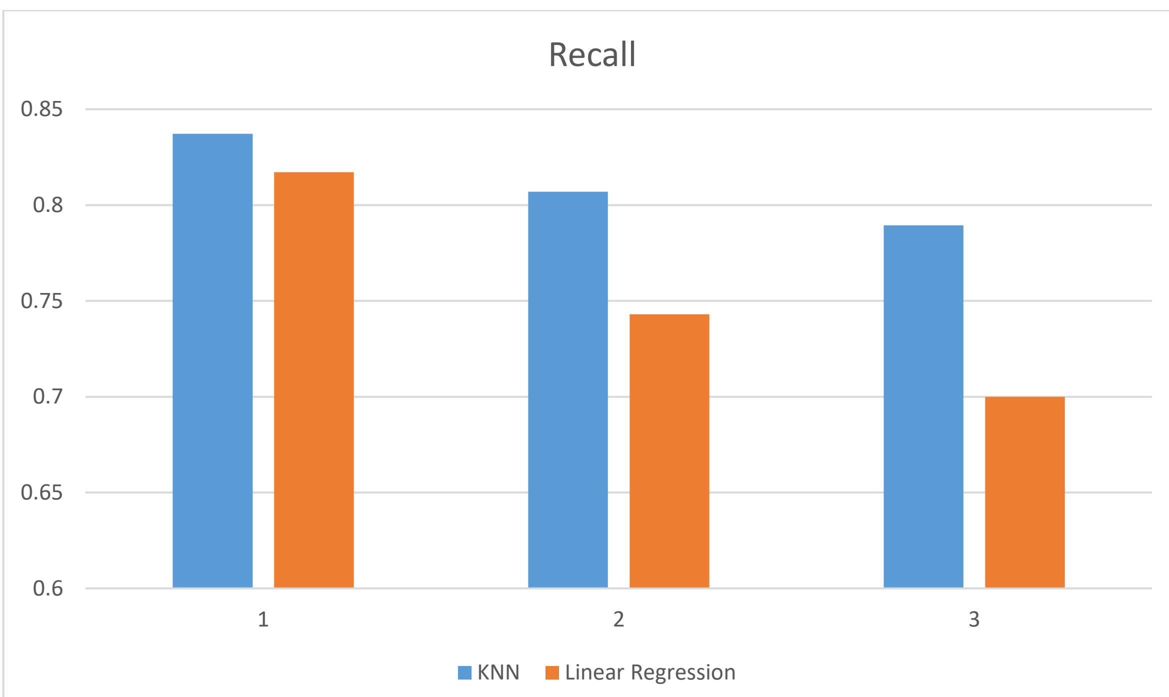

Figure 4-8 Recall changes for three successive data set

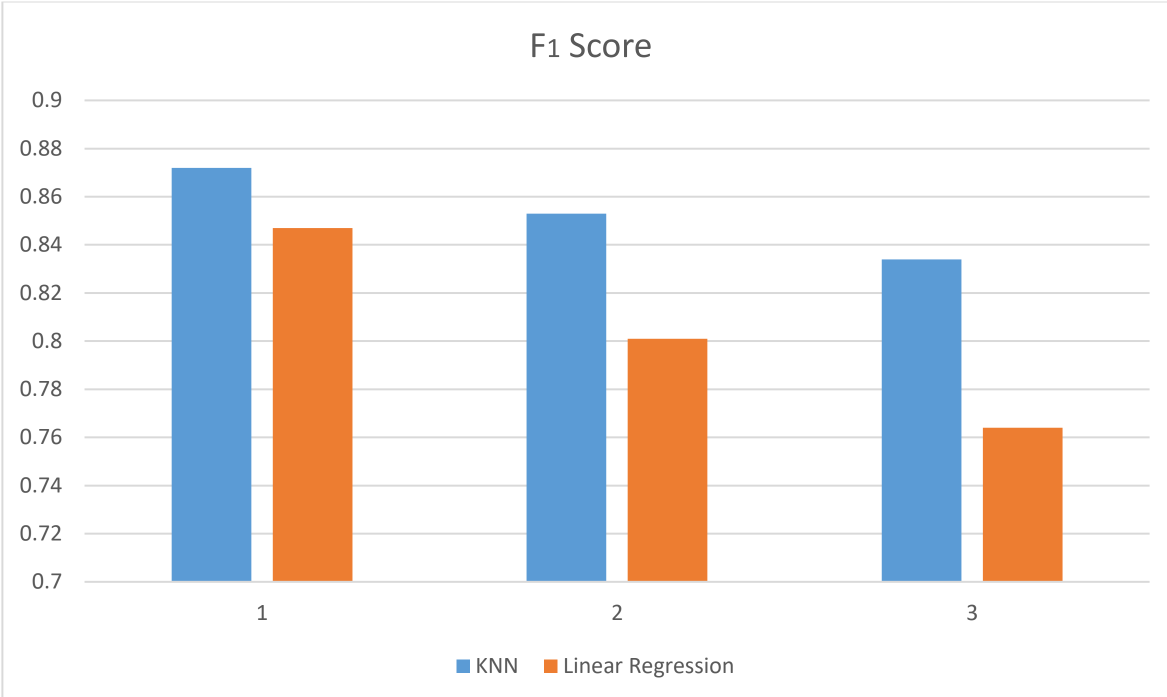

Figure 4-9 F1 score changes for three successive data set

From the results of figures, we can conclude that both algorithms may suffer performance degradation as more data are collected, but KNN is more stable than linear classification as time passing by. The main reason is the unbalance in collected data have introduced bias into the algorithm (unbalanced fitting which is discussed in introduction part), and the model may learn from these bias as the new data is stored as data base. The figure shows that recall has dropped fast for regression and lead to a drop of F1 score. This means the algorithm's ability to predict small cell has dropped. Since linear classification will generate the decision boundary line according to the whole data base. The initial unbalance may lead to a biased line (like UEs are collected nearer to small cell). As the collected data is increasing, the bias may be more severe if the collecting method stay the same. KNN, however, will not take the whole data base as the reference. Therefore, the unbalance of collecting data may not affect KNN severely. As a result, we decide to apply KNN to establish the SON system to monitor the new data set, whenever F1 score is below 0.8, rebalance the data and get new CRE and *alpha*.

So far, we have managed to use machine learning algorithms to find optimal CRE bias to obtain certain offloading objective, and establish SON system to maintain the offloading rate by modifying CRE and edge user factor. The objective for the second part of this chapter is to establish a cache system with machine learning algorithms, so that the small cell backhaul traffic issue can be addressed.

4.4 Application to a Cache System

HetNets have been considered as a solution to meet 1000 times data requirement of 5G generation in the next decade [138]. This promising topology has been proposed to deploy large-scale small-cell networks based on an existing macrocell network. Since small cells will be allocated to a lamp pole or a bus station to get closer to UEs, the advantage is obvious [139]. Small cells can provide better SINR because of low signal path loss, and HetNets propose high spectral efficiency to increase data capacity [4]. However, both features require the operators to provide high-speed and reliable backhaul for small-cell networks. On one hand, a traditional optical fibre may not be the option because the CAPEX cost for large-scale small cells is rather high; besides, the location of small cells indicates that it is difficult to install fibres for each cell. On the other hand, the limited bandwidth for operators is a bottleneck in providing high-speed wireless backhaul for small-cell networks [140].

Under such circumstances, a method to offload backhaul traffic is urgently required. Research shows that frequent downloads of some popular content comprise significant part of mobile data traffic [141]. As a result, if these popular contents can be pre-stored in a local area instead of downloading through the network every time, a significant backhaul traffic can be saved, which is the concept of cache. A cache is a 'hardware or software component' used to store data so that the request of these contents can be made quicker and easier; what contents should be stored can be suggested from algorithm computation or frequent duplicate requests [142]. In HetNets, this concept can be realised to install high-capacity cache storage devices in small cells, where backhaul traffic can be offloaded [143]. The choice of contents may depend on their popularity and certain users may have relatively fixed preference pattern [144]. By controlling the downloading contents and times of small-cell users, the QoS offered to users can be significantly increased.

Cache has been thoroughly studied from traditional optimization method and stochastic geometry. By now, the concept of big data has getting more and more popular, with the help of established data base and data mining method, we can generate practical model according to various scenario. Mobile traffic data is just one of the potential data mining subject, it can reflect human behaviours including routine, preference and so on [145].

4.4.1 System Model

A two-tier HetNets system is considered in this scenario. As mentioned in last section, macrocell are connected to optical fiber for backhaul. Meanwhile, small cell will apply wireless backhaul and cache storage device is installed in particular small cells. $V = \{V_1, V_2, V_3, \ldots\}$ is the set of all possible contents. $R=\{R_1,R_2,\ldots R_N\}$ is set of small cell users with total user number as N. Suppose they have the same request times R during the time T, then each of them has a content set, $Ci=\{C_{i1}, C_{i2}, \ldots C_{iR}\}$, which value is selected from set V. We suppose cache can only be installed in small cell because of limited bandwidth for wireless backhaul.

When users request to download contents which pre-stored in local cache, this part of backhaul traffic will be considered to be offloaded because core network router is not involved in this local transaction.

4.4.2 User Preference Pattern (UPP)

Each user may have specific preference when requesting contents. Some people likes to watch action movie, while others prefer romantic movie. If we quantilize the preference as score and listed in vector, we may obtain the pattern for each user and the basic conditions of applying k-means are reached. In order to test the performance of proposed model, we have selected two typical content categories to apply the model separately: YouTube videos and movie. Each category has his own specific UPP.

YouTube was first launched in 2005 and has become a platform for people uploading videos including online lecture, music, game video and so on. This platform has brought up a popular new idea so that the users keep growing. Even in 2006, the daily viewed video reached 100 million and daily uploaded video exceeded 65 thousand. By the year of 2015, 81% internet users worldwide have already visited YouTube and 31% of the website user are frequent users, which request to visit at least once a day [146]. In YouTube, videos made by same person are normally categorized in the same channel. Figure 4.11 shows the most popular YouTube channels for December of 2017 [146]. For example, Justin Bieber was third popular channel in YouTube with 32.91 million subscribers. Therefore, we can consider the subscribers number as a popularity degree and map the universal UPP for top 20 channels in Figure4.12. After that, we can generate the UPP for each single user based on the universal one, but add random bias according to zero-mean standard Gaussian random variable model on each channel score and has a 95% confidence interval that the bias is between -30% to 30%, so that the diversity of users can be maintained. (we cannot obtain the exact subscription for real YouTube users, because it is the privacy protected by law. Therefore, we can only simulate each user's UPP according to the accessible big data investigation)

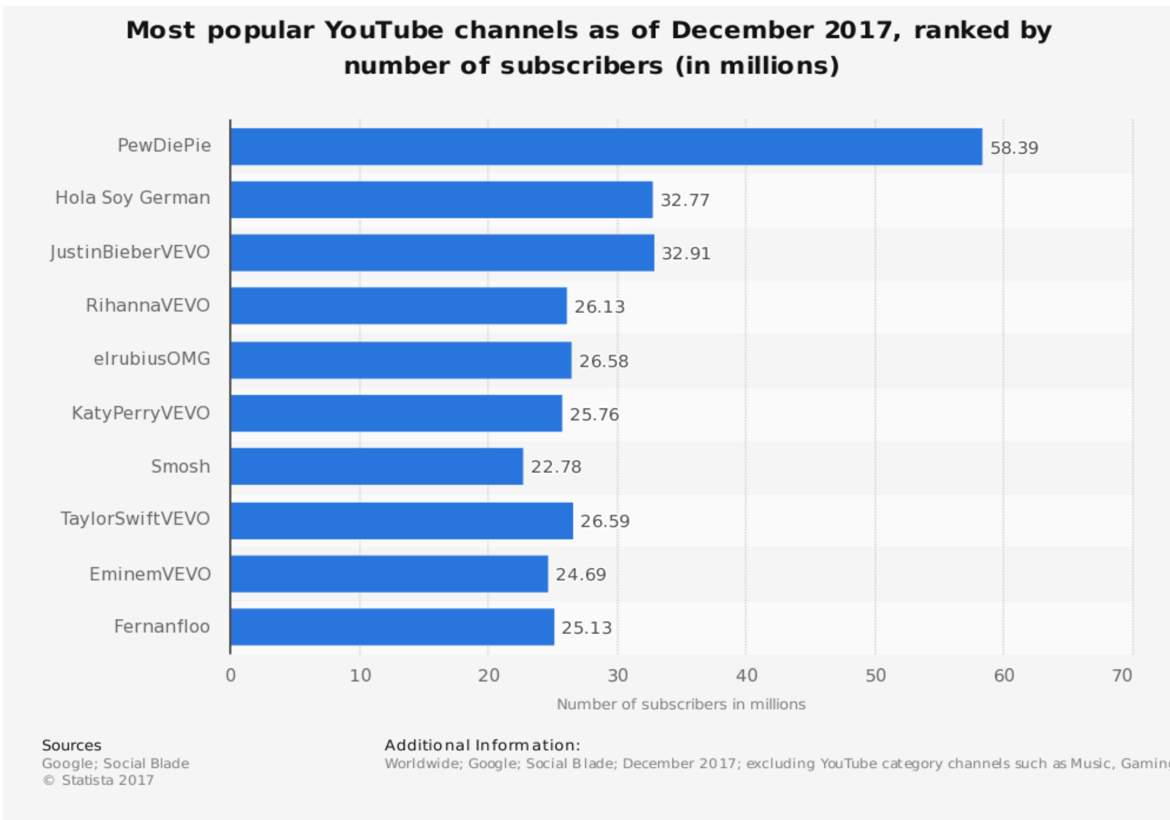

Figure 4-10 Most popular YouTube Channels of Dec 2017

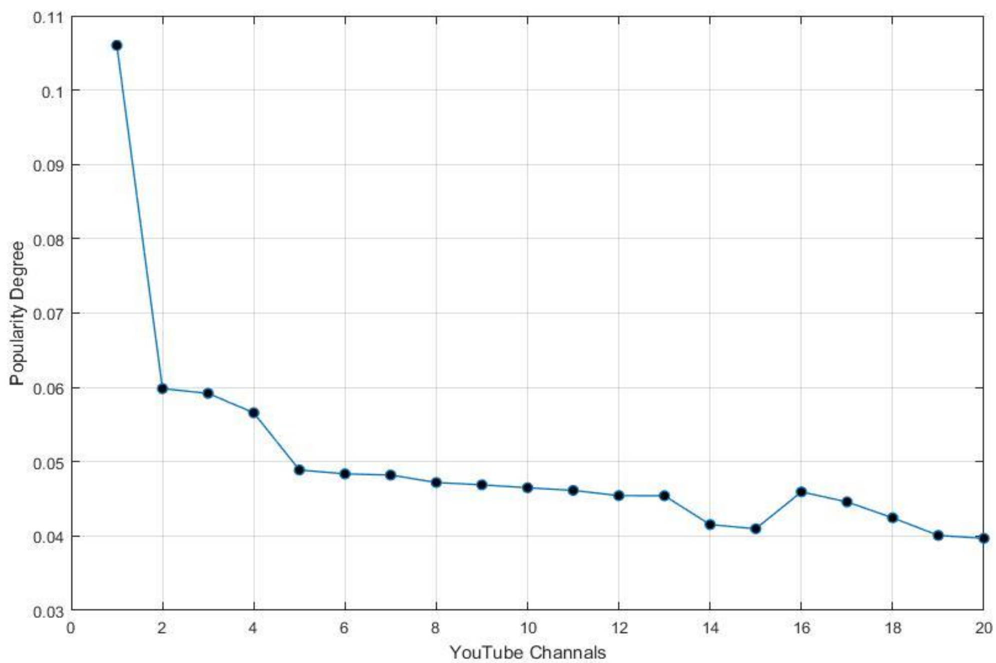

Figure 4-11 UPP for top 20 YouTube Channels

The second typical cache content we concern is movie. Unlike what we do for YouTube that we category contents by channels, we choose to category movie videos according to their 'genre'. Genre is the term meaning the collection of movie types according to their topic, target audience, historical background and so on. In order to map the preference of movie to UPP, we have adopted the survey result of [147], which investigates surveyor's score on different type of movie. During the survey, the author has chosen 17 common movie genres as follows: (1) action movie, (2) adventure movie, (3) animation movie, (4) comedy movie, (5) crime movie, (6) drama, (7) erotic movie, (8) fantasy movie, (9) 'hermit' film, (10) historic movie, (11) horror movie, (12) mystery movie, (13) romance, (14) science fiction movie, (15) thriller movie, (16) war movie, (17) Western movie. The detail of the investigation tries to find out young adults' preference and let interviewers to mark every genre from 0-10 according to their experience. According to the result, we have plotted the universal UPP for movie preference (men) in Figure 4.13.

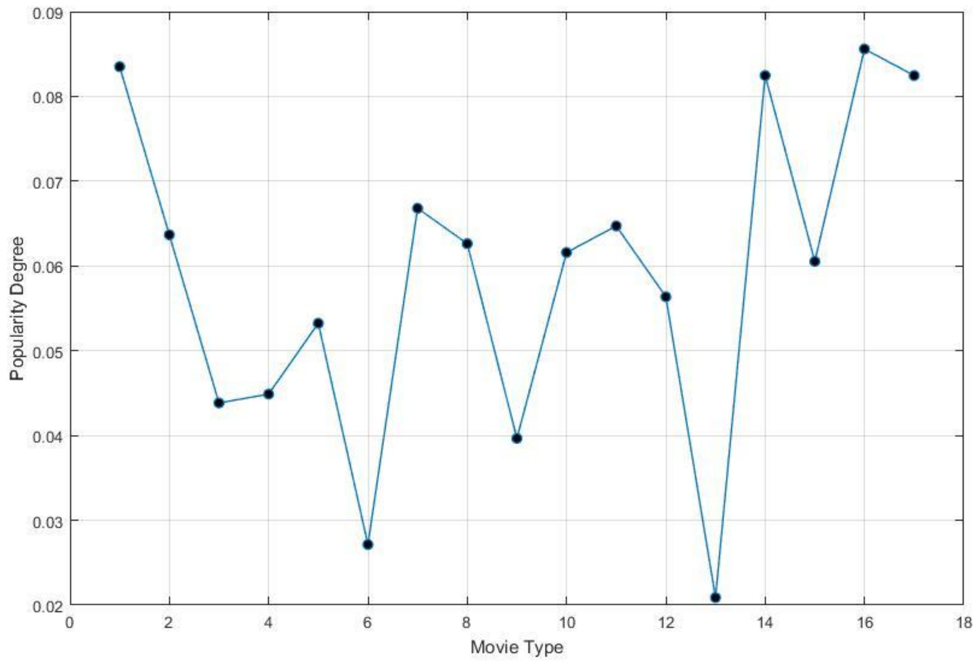

Figure 4-12 UPP for 17 Movie Types

4.4.3 Cache Hit-ratio

In order to evaluate the established cache memory system, we can calculate the hit-ratio of the system when a set of users are applied. If the hit-ratio is high, it means that more users' requests are stratified by cache and no need to access backbone network. In other words, more data traffic is offloaded from the main stream if hit-ratio is high, which is what we want. The formula of hit-ratio is shown below:

$$H = \frac{\sum_{i=1}^{N'} \sum_{j=1}^{R} a_{i,j} C_{i,j}}{\sum_{i=1}^{N'} \sum_{j=1}^{R} a_{i,j} C_{i,j} + \sum_{i=1}^{N'} \sum_{j=1}^{R} b_{i,j} C_{i,j}} \qquad (4.10)$$

Where H is hit ratio, N' is total user number who access cache, R is number of request, $C_{i,j}$represent the content, $a_{i,j}$ is hit indicator and $b_{i,j}$ is miss indicator, which are either 0 or 1. If $C_{i,j}$ is stored in cache, it is 1, which means cache memory hits this content; otherwise it is 0, which means cache memory misses this content. And then average UE's request time from cache is calculated as follows:

$$T' = cH + m(1 - H) \tag{4.11}$$

Where c = time to obtain content from cache, and m means the time penalty for miss content from cache.

Since we aim to increase offloading factor to reduce UE's accessing time and small cell network backhaul traffic with limited cache size, *S* (in this paper we set maximum cache size over total content is lower than 50%), the problem can be formulated as follows:

$$\max_{a_{i,j}} \; H(a_{i,j}) \tag{4.12}$$

$$s.t. \quad a_{i,j} \in \{0,1\}, i \in N', j \in R$$

$$C_{i,j} \in V$$

$$\frac{S}{V} \leq 50\%$$

4.4.4 K-means Algorithm and KNN Algorithm application

The last part shows that, if we manage to increase Hit-ratio of the cache system, we will be able to offload more data traffic from small cell network, so that the backhaul issue of HetNets can be mitigated. In previous sections, we have quantified each user's preference into a UPP (either according to their actual watching behaviour or their score on various types of content through investigation). With the collection of all users' UPP in data set, we may analyse and summarize the property of each user and partition users with similar behaviours into several groups. According to the centroid of each cluster, we will be able to proactively store contents in respect small cell which satisfies this group of users. After transferred the necessary information of the scenario into quantized data, we have managed to apply K-means Clustering Algorithm in this work and hence establish the cache system.

In order to maintain a high QoS for small cell users, we aim to reach two quality standards. First one is the overall satisfaction rate. If the user's request is satisfied by the cache system, it means this user will enjoy low latency because this request will not go through the whole HetNets to backbone network; it also means this user will enjoy high downloading data rate because it is served by small cell cache storage, which is supposed to be near to him and specially designed to provide the requested contents. As a result, we use the average hit-ratio of all users to present the overall satisfaction rate. Secondly, the QoS of each single user should also be maintained. If the user has low hit-ratio, it will not only suffer long latency because of the void request time to cache system but average low data rate compared to other better cache-served users. Suppose the cache system store the contents randomly and has the cache limit of 50% of all contents. The probability of one random user's request of content being satisfied by cache system should be 50%. Under such condition, if one user has hit-ratio lower than 50%, it means the cache system performs bad for this user, which cause unsatisfactory. In conclusion, we should build the cache system with both high overall satisfaction rate and low unsatisfied users, and the modified K-means Clustering Algorithm is shown below:

Algorithm 4.4: Cache System Design with KCA

Input:

- Training Data set *R* with total element number N_1, each element within has *v* parameters V_1, V_2. … V_v
- User content Matrix *C*, where row means index number of user and column means index number of requests, the value of C is the type of contents.
- Initial number of cluster, *K=1*
- Limitation of cache storage *L'*
- Set iteration step variable *i* =1, *j*=1

Initialization and Iterations

1. Normalize data set *R* into UPP data set, new parameter $Vi' = Vi/\sum_{j=1}^{v} Vj$
2. **for** i = 1 to N_1 **do (**loop for all elements within training data)
3.   **for** j = 1 to *v* **do** (loop for all parameters, such as Youtube channels or movie genre)
4.   Generate user content matrix *C*
5. $C_{i,j}$ = generated number of requests according to current user UPP.
6.   **end for**
7. **end for**
8. **while** $K \leq$ *limitation of clusters number* **do**
9.   Randomly select k elements as initial cluster centroids (c_1, c_2, … c_k)
10.   **for** i = 1 to N_1 **do**
11. I_i = index of cluster whose centroid has the minimum Euclidean distance to current element, R_i
12.   **end for**
13.   **for** j = 1 to *K* **do**
14. μ_j = mean of all elements assigned to current cluster, $G_j = \{ R \mid I_i = j\}$
15.   **end for**
16. Calculate error for current partition $E' = \sum_{j=1}^{K} \sum_{i \in G_j} ||R_{i,j} - \mu_j||^2$
17. **if** $|E'-E|>0$ **then**
18. $E:=E'$
19.   Return to step 9 with new cluster centroids as (μ_1, μ_2, …μ_k)
20. **end if**
21. **for** *L* = 1 to *L'* **do**
22. Set up cache on each cluster with top *L* parameters as cache contents
23. *H* = average Hit-Ratio for whole user content matrix *C* with current clusters partition
24. *F* = rate of users with Hit-Ratio lower than 50%
25. **if** $H \geq 70\%$ *and* $F \leq 5\%$ **then**
26.   break while and return *K*, I_i, *L*
27. **end if**
28. return to step 7 with *K=K+1*

29. **end while**
30. **return** K, $I_{i,}$, L

Once the cache system is established, the next step is to implement the SON system so that any new user entering into the cache system can be automatically assigned to its suitable group and receive cache service from the suitable small cell cache server. With the cluster cache system constructed by KCA, we can continue to use KNN classification method to auto-assign users from new data set, where the cache system can be considered as the pre-stored training set for this supervised algorithm. The detailed algorithm is shown below:

Algorithm 4.5: Partition and SON Design for existing Cache System

---

Input:

- Data set R with total element number N_1, each element within has v parameter types V_1, V_2. … V_v
- Test data set R' with total element number N_2, each element within has v parameters V_1, V_2. … V_v
- Initial user content Matrix C_1, where row means index number of user and column means index number of requests, the value of C is the type of contents.
- Limitation of cache storage L'
- Number of considered neighbour, $K'=5$
- Set iteration step variable i =1, j=1

Initialization and Iterations

1. Predefine the value of K' according to the scenario
2. Apply algorithm 4 to obtain Cluster index set $I = \{I_1, I_2, \ldots, I_i\}$ as classify label set
3. Combine training data set R with index set I as stored training set for KNN algorithm
4. Normalize test data set R' into UPP data set R', $Vi' = Vi/\sum_{j=1}^{v} Vj$

for i = 1 to N_1 **do** (loop for all elements within training data)

  for j = 1 to v **do** (loop for all parameters of each element)

  $C_{i,j}$ = generated number of requests according to current user UPP.

  end for

end for

for i = 1 to N_2 **do** (loop for all elements within test data)

  for j = 1 to N_1 **do** (loop for all elements within training data to find nearest ones)

   $D_{i,j}$ = Euclidean distance between the i_{th} test data R_i' and the j_{th} training data R_j

  end for

  Sort current set of R and obtain the top K' elements having shortest value

  $VOTE$ = set of number of votes on cluster within the top K'

  find highest vote number within $VOTE$ set

   R'_i is classified to the cluster with highest vote number

  end find

end for

H = average Hit-Ratio for whole user content matrix C with current clusters partition

F = rate of users with Hit-Ratio lower than 50%

if $H \leq 60\%$ *or* $F \geq 10\%$ **then**

  Return to Algorithm 4

else if

  return H, F, move to next test data set

end if

4.4.5 Recommendation system

So far, we have supposed that our collected data are sufficient, which means that the UPP for each user is complete. However, during the phase of collecting data to establish UPP, either the data for specific content type is not counted or user does not provide the score for specific content type (Si-fi movie or Justin YouTube channel). As a result, UPP for new users may not be complete and we cannot distribute the new users to his suitable cache server. In such situation, we can use our proposed cache system to predict the missing score in UPP according to existing score and established clusters, even if we are still not sure which group this user belongs to. This is the second application of our proposed cache system. Equation

(13) will calculate the similarity between missing score (*u*) and given scores (*m*). With the similarity of all given scores, we can use Equation (14) to predict the missing score.

$$sim(u,m) = \frac{\sum_{i=1}^{V}(P_{ui} - \bar{P}_u)(P_{mi} - \bar{P}_m)}{\sqrt{\sum_{i=1}^{V}(P_{ui} - \bar{P}_u)^2}\sqrt{\sum_{i=1}^{V}(P_{mi} - \bar{P}_m)^2}} \tag{4.13}$$

$$P_{ui} = \frac{\sum_{m=1}^{k} sim(u,m) \times (P_{vi} - \bar{P}_v)}{\sum_{m=1}^{k}|sim(u,m)|} \tag{4.14}$$

4.4.6 Simulation and Analysis for Cache System

In order to evaluate our proposed cache algorithm, the first part of simulation is to find out optimal K and respect central pattern and then evaluate the performance. During the simulation, we suppose the number of UEs for both scenarios (YouTube and Movie) is 1000 (*N*=1000), and suppose each UE has 1000 times of request of contents (*R*=1000), so that the total request number during this period is 1,000,000 times. As mentioned in last section, our objective is to realize at least 70% traffic is offloaded by our cache system with only 50% cache size. Meanwhile, user's unsatisfactory rate (UE with hit-ratio lower than 50%) should be maintained under 5%. And then we apply Algorithm 4 on two scenarios.

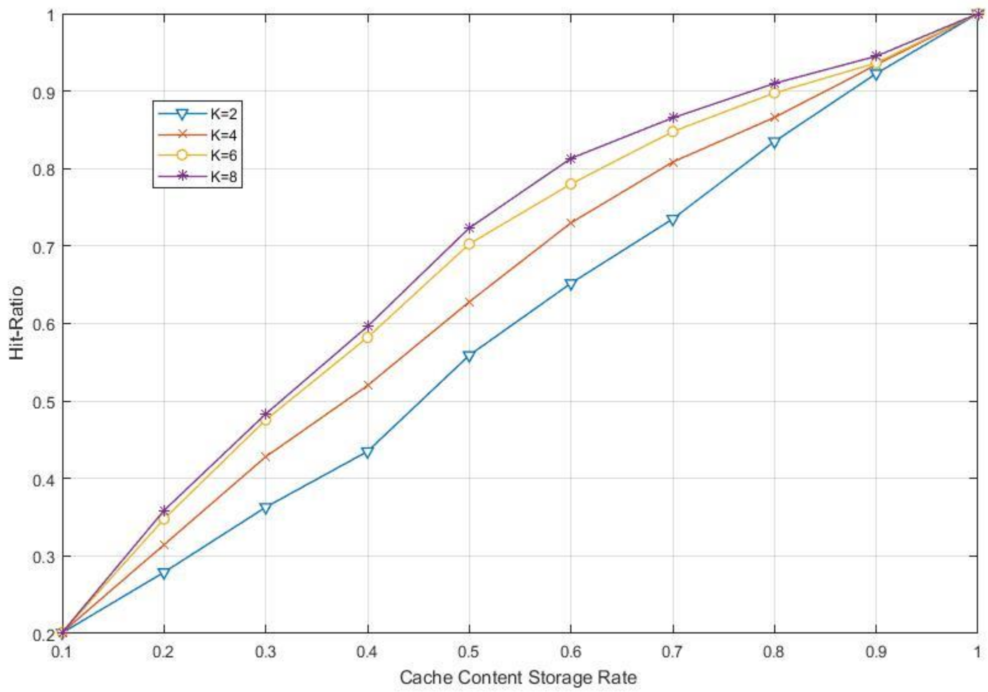

Figure 4-13 Hit-Ratio for different K number on YouTube

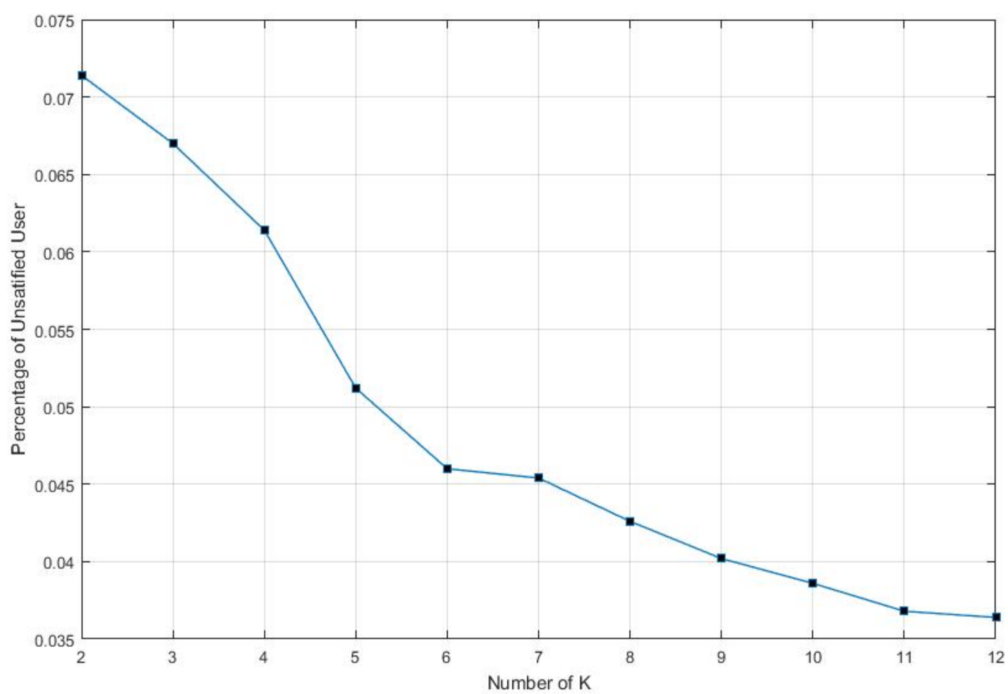

Figure 4-14 Percentage of Unsatisfied User changes with K number for YouTube

Figure 4.14 shows the offloaded traffic from small cell network (can be also considered as Hit-ratio) changes with cache size under different K values for YouTube scenario. We can observe that when cache size reaches 50%, models with K larger than 6 will satisfy the predefined 70% offloading requirement, and higher K value will help to enhance the performance of the cache system. Nevertheless, the effects may gradually diminish as K increases. As K increases, more small cells will be introduced to behave as cache servers, and each of them will have its own unique pattern so that UEs with similar UPP will choose it as cache server. Higher K value means more options for UEs, so that they can choose better centroid and hit-ratio will rise accordingly. 'Outlier' users, however, may have UPP highly unlike traditional one, and the effect of more centroids is limited for these outliers. Figure 4.15 shows percentage of UEs, which have hit-ratio lower than 50%, changes with K value. From K= 2 to 7, the percentage drops fast but the effect is gradually decreased and the percentage almost remains the same after K=9. This will also explains one phenomena of Figure 4.14. We can observe that form K=2 to K=6, the Hit-ratio has a dramatic increase, thanks to the diminishment of 'outliers'. However, from K =6 to K =8, the margin benefit drops, so that plot K=6 and K=8 are much closer. This is because the remaining 'outlier' is harder to be eliminated, which is shown on Figure 4.15.

With the increase of K, the outlier may eventually disappear. However, this does not mean that K should be as large as possible. On one hand, larger K means we need to set up more small cell cache devices, and the CAPEX will increase significantly. On the other hand, the margin effect of K will also get lower with the increase of K. Therefore, we need to set up an upper bound of number K to maintain a balance. For this scenario, we suppose each small cell should serve at least 100 UE to maintain a sufficient load, which means the maximum K number = 10.

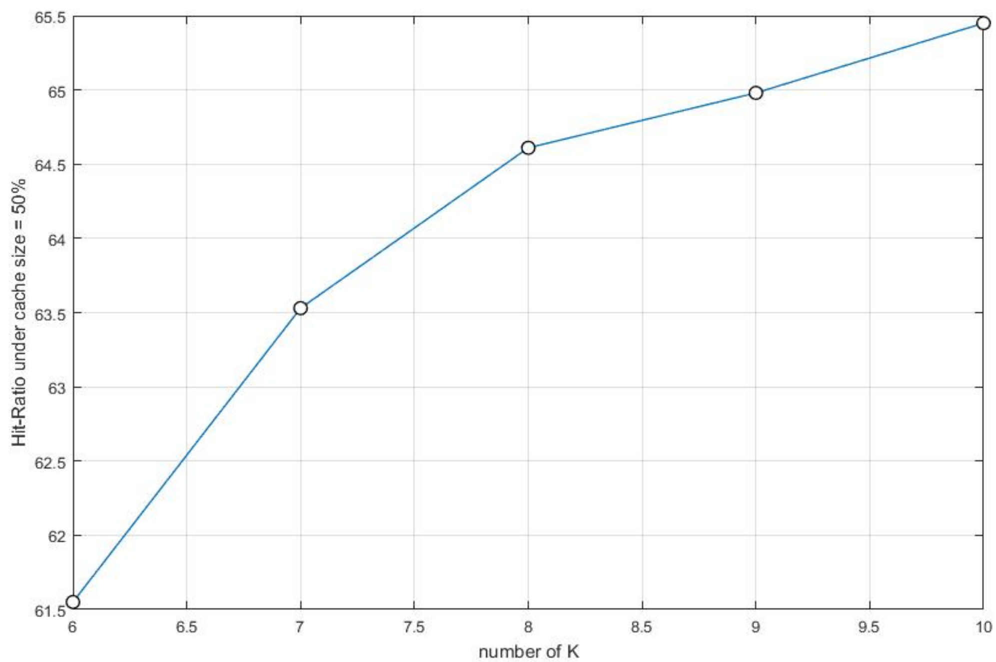

Figure 4-15 Hit-ratio under 50% cache size changes with K for Movie type

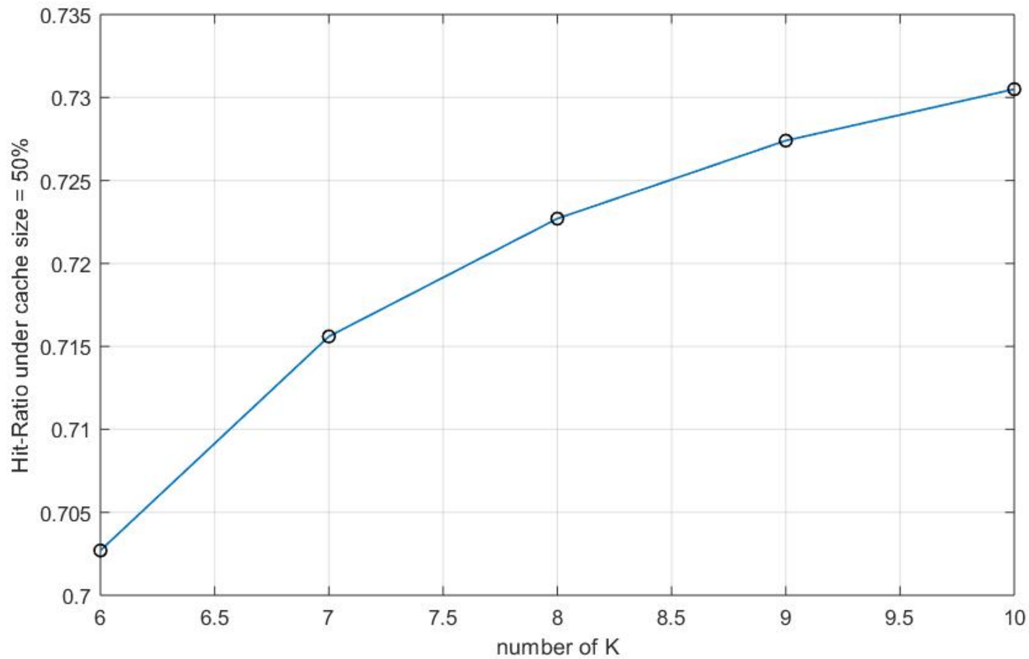

Figure 4-16 Hit-ratio under 50% cache size changes with K for YouTube

Figure 4.16 shows the Hit-ratio of system under 50% cache size (actually, it is over 50% because we choose 9 over 17) changes with K for Movie scenario. It is obvious that even if K reaches the upper bound, Hit-ratio still fails to reach minimum requirement. Figure 4.17, however, clearly shows that the Hit-ratio has already reached 70.27% for YouTube when K=6. The Hit-ratio for YouTube continues to increase to 73.05% when K =10. Moreover, both graphs has shown that the effects of increasing K is gradually decreasing, which indicating that the number of K should also be restricted to maintain efficiency. The reason for these phenomena may be the difference between two scenarios' universal UPP. We will detailed discuss how UPP and number of K affect cache system in the following part.

As a result, by considering predefined constraint and the result from two figures, K=6 is the optimal value for the YouTube scenario with traffic offloading rate reaching 70.27%, and

K=10 is the optimal value for the movie scenario with traffic offloading rate reaching 65.55%. Figure 4.17 shows the UPP for 6 cluster centroids in YouTube scenario, which is the 'example' for UEs bound to it. Figure 4.18 shows 10 cluster centroids in Movie scenario. For the small cell cache device representing the respect centroids, it will store the top 50% cache contents according the popularity degree score, so that the small cell cache system is implemented.

Now, we will compare these two optimal centroid graphs to find out what leads to different performance in two scenarios. It can be observed that the centroids in movie scenario is more similar to each other than that in YouTube. In YouTube universal UPP, the score for all channel types generally have no huge difference except for the first channel (PewdiePie). In movie one, however, there are 4 very popular movie type, and 3 very unpopular movie type. As a result, most of users will follow this pattern and the top 4 movie type will be chosen for every cluster and 3 'poor' movie type will be discarded by every cluster. Under such situations, the outlier that have high request for these movie contents may hardly to be satisfied unless K is very high, so that they may have spare device to cache these contents. This is why the cache system performs poorer in movie scenario. As a result, the choice of candidate contents is also very important during implementing small cell cache system. We should try to make more contents but less difference in the candidate pool; more content types will increase the diversity of centroids (this requires large data collection, highly depends on our accessible resources) so that outlier may be also taken care of; less difference means that contents with low popularity degree should not be added into the candidate poor, so that the system will not waste time considering outliers. The second objective requires the system has a SON to keep monitoring the change of popularity as more data are analysed.

Once the performance drops, the system may check the new candidate pool, and redesign the cache system. This can be done by our proposed algorithm 5.

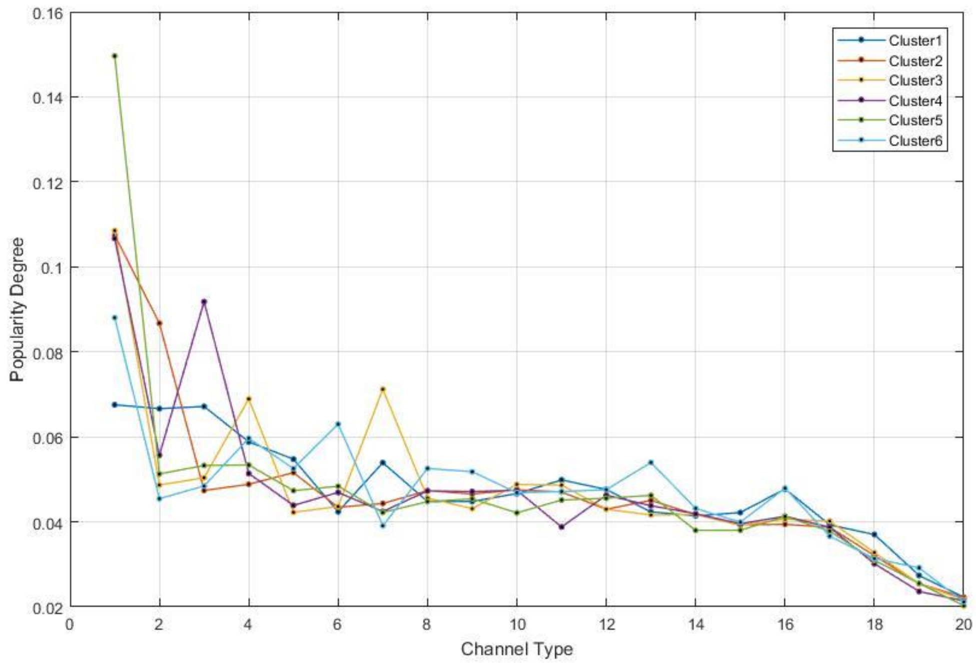

Figure 4-17 UPP for 6 cluster centroids in YouTube scenario

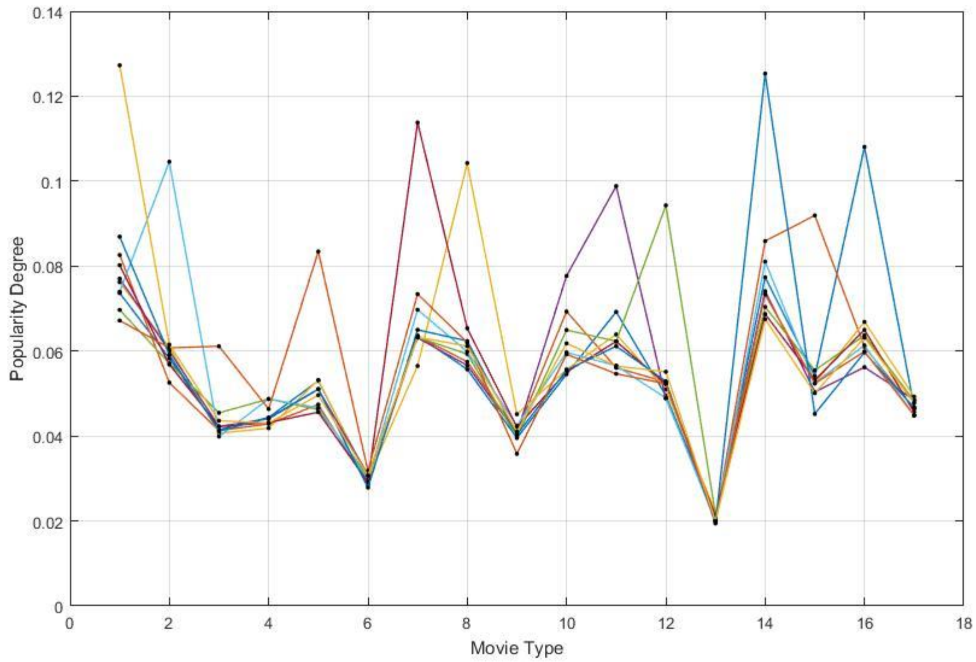

Figure 4-18 UPP for10 cluster centroids in Movie scenario

After establishing the cache system with proposed algorithm, our second step is to apply KNN method with existing clusters and evaluate this cache system in new data set. We generate 5 test data sets on YouTube, which contains 500 UEs with UPP in each data set because we are now trying to classify them to different small cell cache servers. In order to evaluate the cache system for new UEs, we suppose each UE request 1000 times during the period and calculate the hit-ratio. Figure4.19 shows the hit-ratio for all new UEs and Figure4.20 shows the average hit-ratio and percentage of unsatisfied UEs for 5 data sets. We can observe that the hit-ratio is still maintained at high level varied from 67.94% to 69.21%, and 4 of 5 data sets' unsatisfied rate are below 5%. Therefore, we can conclude that the cache system established with our proposed algorithm performs well not only in clustering existing UEs without UPP but also in classifying new UEs entering the system.

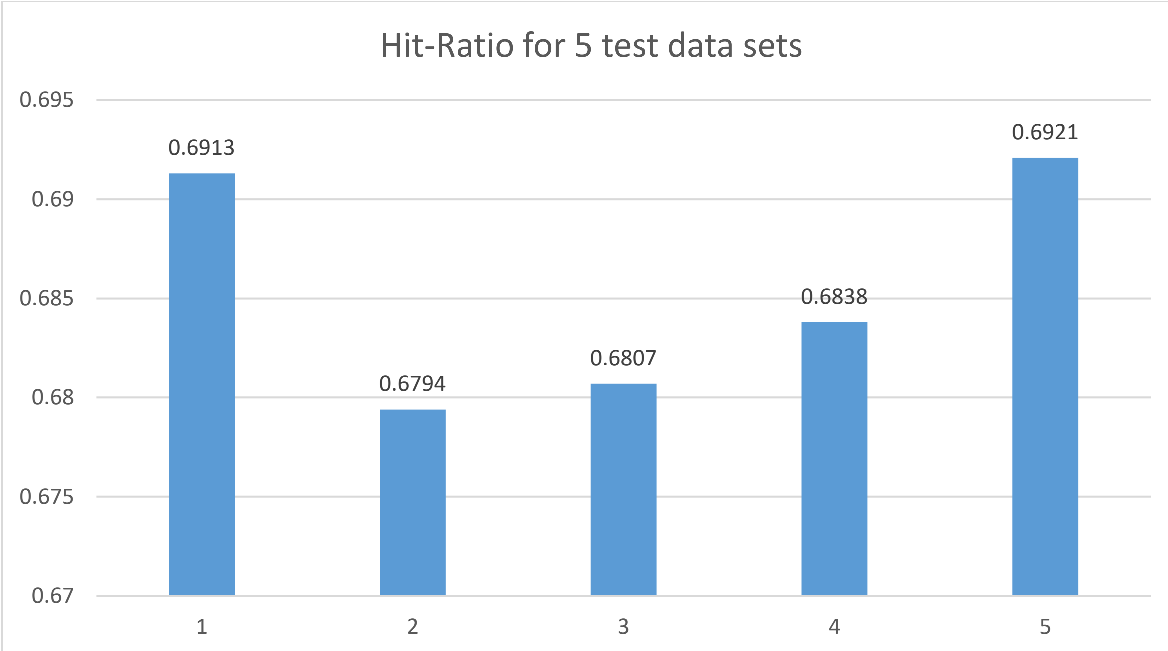

Figure 4-19 Hit-ratio under 50% cache size for 5 new data sets one YouTube

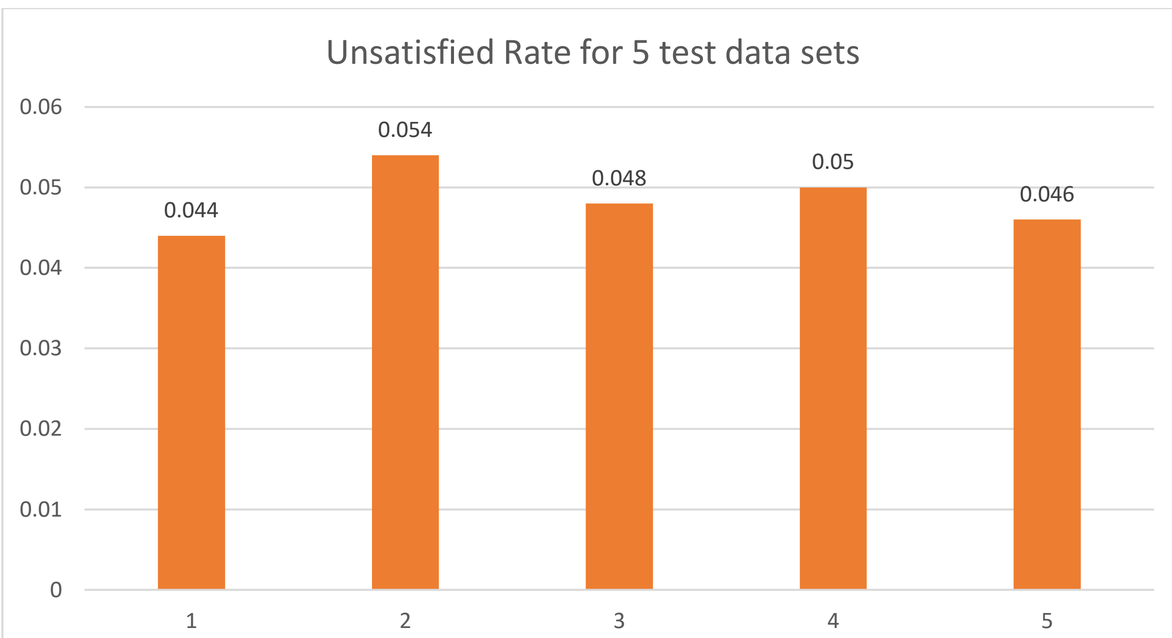

Figure 4-20 Unsatisfied rate under 50% cache size for 5 new data sets one YouTube

Unlike our first design in offloading UEs in HetNets, the ‘traffic map’ now is not updated in real-time. When we want to establish the cache system, what we want to serve is the stable

UEs in the system with constant and frequent requests. Therefore, new UEs who entering the system will not be counted in the 'traffic map' which is used to design cache clustering. Until this UE is proved to be another stable UE which may stay in the system for several successive days and has frequent requests, it will be added in the 'traffic map' to affect the design of cache. Besides that, user's preference may change with time passing by, and will reflect on his requested contents. As a result, the hit-ratio and unsatisfied percentage will be monitored maintain a qualified cache service. Once the value of these two parameters reach the threshold (hit-ratio lower than 60% or unsatisfied UE over 10%), a new cache design will be established based on current 'traffic map' and proposed algorithm, until the system is stable again.

We have discussed the benefit of high Hit-ratio for cache system in last part. Another major advantage of cache system is to reduce latency of content request. Even if we ignore the high DL data rate from small cell, cache system with higher hit-ratio still can reduce latency in terms of less penalty time of missing request. If the average request-respond time between UE and a cellular cell is 20 ms [151] and we assume that the penalty request time from small cell UE to backbone network will contain relaying at least two times (UE – small cell – macro cell) and plus at least 100ms waiting time due to backhaul congestion. By applying (4.11), we can find out the latency difference among no cache system, 10%-50% cache size with proposed optimization and 50% cache size system (without optimisation). The result for Youtube scenario is shown in Figure 4.21. The first one is the latency time without cache system, which is 140 ms. Our objective is to apply cache system to reduce latency. The last one is the conventional cache system with 50% cache size, but there is no optimization applied. The latency has been reduced to 80 ms. From 2 – 6 is the latency when our proposed cache system with different cache size is applied. The results show that the proposed cache system can reach the same effect of conventional cache system with only 30% cache size. If we use the same cache size of conventional cache system, our proposed scheme can reduce the latency by up to 60.21% and reach 55.68 ms.

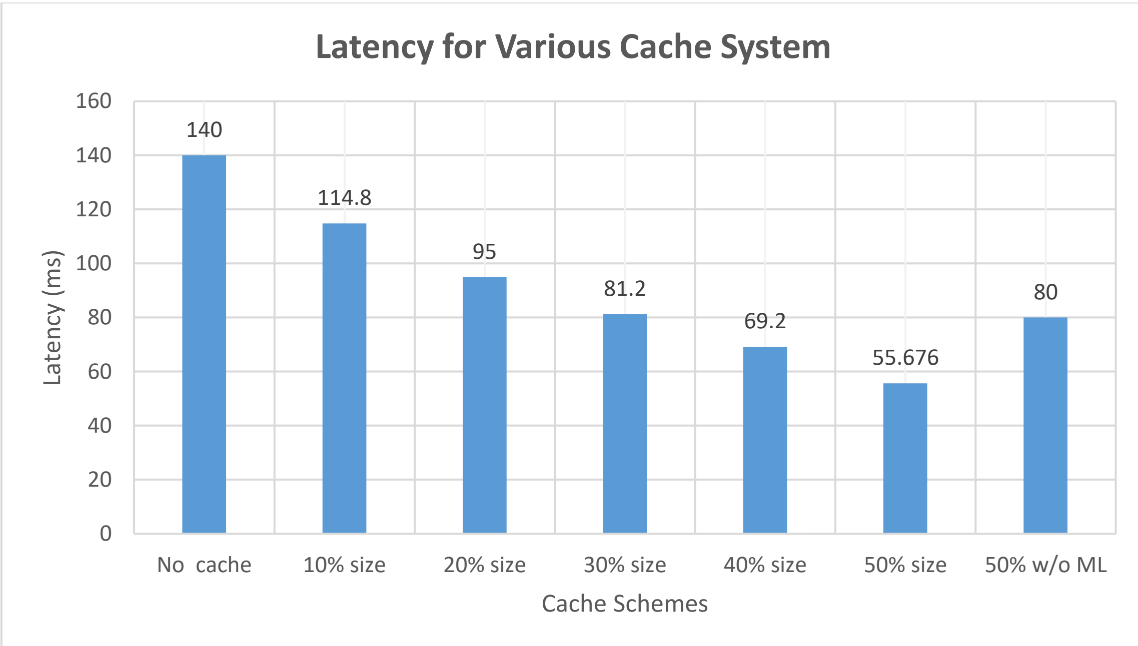

Figure 4-21 Latency for Various Cache System

Besides improving Hit-Ratio and latency situation, our proposed cache system may also provide another useful application, which is the prediction of user's popularity score on specific channel if provided UPP is not completed. For example, a new user entering the system and he only scores on several channels so that we cannot obtain complete UPP. We also cannot accumulate enough content request to plot UPP because he is new to the system. As a result, we cannot classify this UE into his suitable cluster without UPP. Under such situation, we can use our proposed algorithm in 4.4.5 and established cache system to predict his missing score on UPP. In order to test the algorithm, we generate a new data set with 100 UEs but their score on TaylorSwift are hidden to the system. And then we predict the score of this channel with our proposed algorithm, through finding 5 most similar UEs in the system according to remaining 19 scores. Figure 4.21 shows the comparison of hidden score with predicted score with algorithm. The original average popularity degree score on Tayler channel for this data set is 0.0497, and the predicted average popularity degree score on Tayler channel is 0.0482. Both the graph and calculation implies that the two plots generally follow the same pattern, the main reason causing disagreement is the correlation between calculated channels and predicted channel. If the correlation is high and more related channels are calculated, the predicted result will be more accurate. The correlation among YouTube channels or movie types is not the scope of this paper, our objective is to propose a method to implement UPP with limited information.

However, the real user data may not perform as good as our simulated user data. Because of lack of actual data on real users' UPP (protected privacy), we have randomly simulated each user's UPP according to the statistics investigation. The correlation among different channels have been shown according to large data statistics, which ensures the high accuracy of our prediction system. For real data set, the correlation may be affected by many factors, such as region, language, gender and so on. If we collect enough data, we may add new factor into Equation (13) to modify our prediction system.

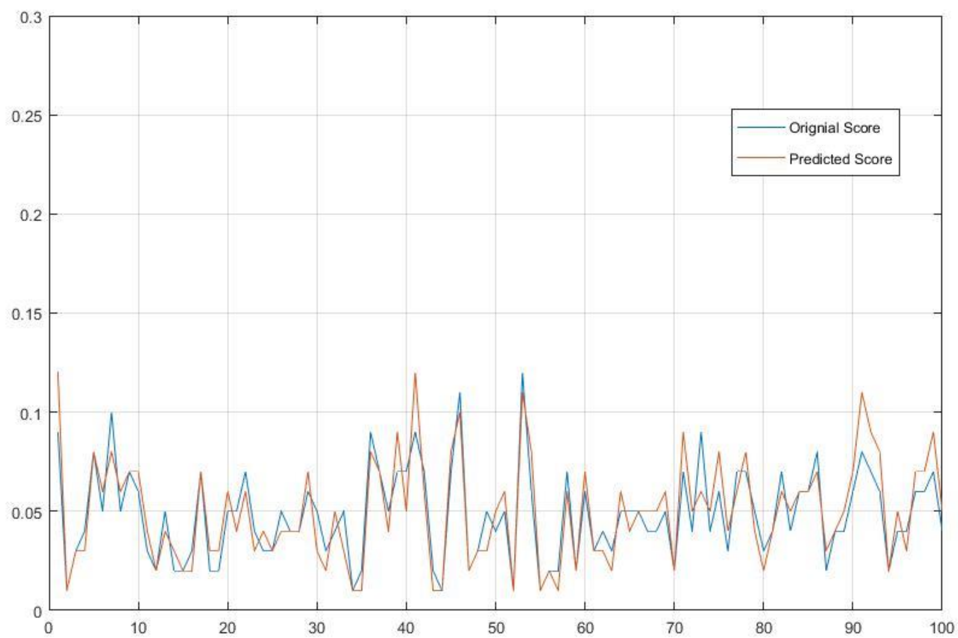

Figure 4-22 Comparison of predicted missing score with original score

4.5 Conclusion

Chapter 4 have managed to solve offloading problem of HetNets, which includes (1) UE offloading from higher tier to lower tier, and (2) small cell backhaul traffic offloading. This chapter applies a widely used unsupervised Machine Learning (ML) algorithm, K-means Clustering Algorithm (KCA) to address these two offloading issues. For first issue, we propose a User-Based K-means Algorithm (UBKCA) by involving HetNets background and Enhanced Inter Cell Interference Coordination (eICIC) to decide the optimal Cell Range Expansion (CRE) bias given specific offloading objective. The center user group set is established to reduce computing complexity. Meanwhile, CRE bias and Edge User Factor are optimized to enhance user offloading so that loading balance objective can be achieved. Simulations are then performed to show UBKCA's better performance than KCA; the optimal combination of CRE bias and Edge User Factor are taken based on both accuracy and offloading factor; furthermore, we have implemented a close-loop SON system with KNN and linear classification so that new UE will be automatically assigned to suitable network tier and offloading factor is maintained within a moderate range. In order to solve the backhaul traffic offloading, we have managed to establish a cache system within small cell by applying modified KCA. With the help of the cache system, both small cell users' download speed and request time will be enhanced. KNN is then applied to predict new users' content preference and prove our cache system's suitability. Besides that, we have also proposed a system to predict users' content preference even if the collected data is not complete.

Chapter 5. Analysing CRE effect on Ping-Pong Handover mitigation during Offloading Process

5. 1 Introduction

During last Chapter, we have introduced one of the eICIC technique - CRE, which is a virtual bias added to RSS from small cell to help offload UEs from macrocell. Although the introducing of CRE in HetNets is not originally designed to mitigate ping-pong Handover rate, its existence does follow the second method mentioned in introduction part. So far, the study of CRE' negative effect has focused on increasing cross-tier interference for small cell edge UEs after offloading them from macrocell. However, how CRE affects UE capacity during offloading phase or Handover phase is not focused yet. Therefore, the objective of this paper is to analyse and evaluate CRE's capability of addressing ping-pong Handover issue, and try to find optimal CRE by considering both negative and positive effects on capacity from Handover aspect.

Besides CRE, there is another value which may affect the bias added to RRS, which is called Handover Hysteresis Margin (HM). HM is specifically applied for the purpose of redundant Handover reduction, which is normally a constant variable that added on serving cell. As a result, both HM and CRE will contribute to the virtual bias. We have mentioned that we need to take UE mobility model into consideration when analysing ping-pong Handover. During simulation, it is hard to record each single UE's exact moving projection and fading situation without actual collected data. In simulation, we will normally introduce mobility model and shadow fading to simulate UE's situation, which may involve random factor and time factor. As a result, a proper model which can predict the status of UE changes with time in statistics should be established, which is the Markov Chain Process (MCP). With the help of MCP model, we have managed to analyse how TTT and CRE help to mitigate ping-pong handover (which are the two methods of mitigating unnecessary handover mentioned in introduction chapter), and optimal CRE value can be obtained given fixed TTT. Furthermore, we have been able to discuss the different effect of CRE and HM under the MCP model. The optimal combination of both parameters has been provided in the end.

5. 2 Methodology

5.2.1 Handover model with MCP

A Markov chain is a stochastic model describing a sequence of possible events in which the probability of each event depends only on the state attained in the previous event [148]. The basic property of this stochastic process is 'memorylessness', which means we can predict the situation of the system in the future according to current situation only, even without knowing the history of the whole process. MCP has been widely applied in establishing the mathematical model for real-world process, which contains random objects and time factors, such as predicting the customers arriving the specific item's arrival sequence, the price of stock and growth rate of observing species [149] – [150].

After analysing how MCP works, we will be able to model Handover process with MCP. In introduction section, we have stated that a complete Handover contains two phases: initialization and process. Take both Handover from macro to small and from small to macro, there are total four main states: M, S, I, I'. We define M states representing UE is bound to macro cell and undergoes initialization phase. Similarly, S states representing UE is bound to small cell and undergoes initialization phase. Besides that, we define I states as handover process from macro to small cell, and I' states as handover process from small to macro cell. During initialize phase, Time-to-Trigger (TTT) is the crucible parameter, which restricts UE from entering second phase unless the candidate's cell's RSS plus virtual bias is larger than current cell's RSS for a period of pre-defined time. In order to map TTT into our MCP model, we divide the whole TTT into several Transmission Time Interval (TTI), and each TTI represents a state within $S = \{S_1, S_2, ...S_n\}$ or $M = \{M_1, M_2, ...M_n\}$. The conventional TTT time may vary from 40ms to 100ms and conventional TTI time is 10ms [152]. As a result, we have adopted 40ms TTT and 10ms TTI for our initial model, so that there will be 4 sub-states in *S* and *M*. We have discussed that longer TTT will further benefit in mitigating ping-pong handover. In our model, this will be represented from two aspects: 1) number of states. Since TTI is defined to be 10ms, longer TTT means more sub-states. With same transition probability, more sub-states means it is harder to jump out of the states chain of *S* or *M*. UE will be less likely to behave ping-pong handover. 2) transition probability. Once the MCP structure of TTT is fixed, CRE will affect the transition probability to increase $P_M(x)$ and reduce $P_S(x)$ to help UE offloaded to Small cell states. As a result, with the combination of TTT and CRE, UE will prefer to be offloaded to small cell and remain stable. The detailed deduction and result will be shown in next part. Since ping-pong Handover represents the frequent serving cell switching between macro cell and small cell, we have modelled the Handover loop as into the Markov Chain with n=4 shown in Figure 5.1.

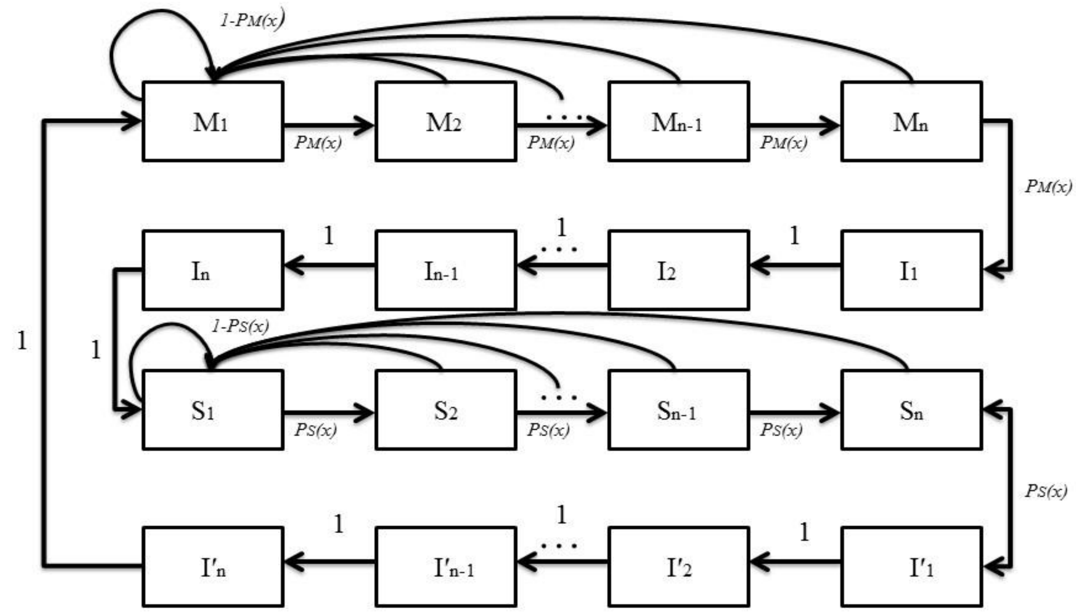

Figure 5-1 Handover model with Markov Chain

After the relationship of all states are defined, we should define the transition probability so that the Markov transit matrix can be established. Suppose UE is bounded to a macro cell and in M1 state, according to UE's location and RSS. It has probability $\mathrm{P_M(x)}$ to reach M2 and $1 - \mathrm{P_M(x)}$ return M1 to renew the Handover assessment and the TTT count time is set to 0. M2, M3, … Mn states follow the same protocol. Until Mn reaches I1, which means the decision condition is satisfied for the whole TTT, the process phase will be started. During this phase, Handover is guaranteed to happen and only control signal and acknowledge signals are transferred to finish Handover process. Therefore, it has probability of 1 moving from I1 to I2 and then to I3 till Handover finishes and reaches S1 states (handover failure is not considered in this paper, so the probability of handover execution phase is set to be 1).

UE will then renew handover process with probability $P_S(x)$ till next handover. During handover process states (I and I' states), mainly traffic signals are transferred and information signal will be blocked till handover finishes. Therefore, frequent ping-pong handover will significantly reduce UE capacity. The virtual bias CRE will take effect to reduce ping-pong handover and increase capacity accordingly, which may be represented in $P_M(x)$ and $P_S(x)$ combined with step number and mobility model. The detailed calculation will be discussed in next session.

After defining MC process and its state transit probability, we will be able to establish the transfer Matrix (*T*) to model the handover process for UEs and analyse how CRE affects it. Table.1 illustrates the 4-state transfer matrix with our algorithm, which will be used in later simulation:

Table 5-1 Markov Transfer Matrix (T)

	M1	M2	M3	M4	I1	I2	I3	I4	S1	S2	S3	S4	I1'	I2'	I3'	I4'
M_1	$1-P_{M(x)}$	$P_{M(x)}$	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M2	$1-P_{M(x)}$	0	$P_{M(x)}$	0	0	0	0	0	0	0	0	0	0	0	0	0
M3	$1-P_{M(x)}$	0	0	$P_{M(x)}$	0	0	0	0	0	0	0	0	0	0	0	0
M4	$1-P_{M(x)}$	0	0	0	$P_{M(x)}$	0	0	0	0	0	0	0	0	0	0	0
I_1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
I_2	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
I_3	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
I_4	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
S_1	0	0	0	0	0	0	0	0	$1-P_{S(x)}$	$P_{S(x)}$	0	0	0	0	0	0

S2	0	0	0	0	0	0	0	0	1-$P_{S(x)}$	0	$P_{S(x)}$	0	0	0	0	0
S3	0	0	0	0	0	0	0	0	1-$P_{S(x)}$	0	0	$P_{S(x)}$	0	0	0	0
S4	0	0	0	0	0	0	0	0	1-$P_{S(x)}$	0	0	0	$P_{S(x)}$	0	0	0
I1'	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
I2'	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
I3'	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
I4'	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The row of Matrix represents ‘start’ state, which means the system is currently in this state. The column for each row means the next state if the ‘start’ state is the row name. Therefore, the value of matrix represents the transition probability from row name to column name. For row M1 to M4, it has probability of 1- $P_M(x)$ to reach the column M1, which means jumping back to the initial state. Meanwhile, it has the probability of $P_M(x)$ moving to next state. As a result, the sum of each row should equals to 1, which means all the possible next states have been taken into consideration. The situation is similar from row S1 to S4. I and I’ row represents the process phase, during which UE is inevitably transferred to the other cell. Therefore, the probability of moving to next state is set to be 1. Still, the total probability of each row is 1 during these two kind of states.

Once transfer matrix is established, the state probability vector V at any given step x can be calculated according to the property of MC:

$$V(x) = V(1)\prod_{i=1}^{x} T(i) \qquad (5.6)$$

$V(1)$ shows the probability of UE at initial point, which is given and can be decided with probability of 1. For example, UE is bound to macro cell from time 1, its probability vector will be [1, 0, 0 ... 0]. With time passing by, number of step will increase, its vector will keep changing according to Equation (6). Therefore, we can find out the vectors for all UEs in the system after x steps and analyse their handover rates through the probability of each state.

5.2.2 *Define Transition Probability*

Since we have discussed that ping-pong Handover may be affected by shadow fading and mobility model. We should analyse the system propagation model to find out how to define the transition probability. Equation (7) is the small cell SINR expression with free path loss and shadow fading. Since we have assumed that there is only one macro cell and one small cell in the HetNets system, the only interference is macrocell RSS.

$$SINR = \frac{P_s l_s g_s}{P_m l_m g_m + \sigma^2} \tag{5.7}$$

Ps and *Pm* represent the transmission power of small cell and macro cell. l_s *and* l_m represent free path loss, and the expression in dB has followed (3.8) in chapter 3. g_s and g_m is the fast-fading gain which is assumed as Rayleigh Distribution. As a result, the expression of *g* should be the exponential random variable. In conclusion, RSS from a cell will be consist of two parts, the distance-decided part $P_s l_s$ and random part g_s. And RSS for macrocell and small cell is shown below:

$$R_m = P_m l_m g_m \tag{5.8}$$

$$R_s = P_s l_s g_s \tag{5.9}$$

According to the discussion in early part, we know that UE will start a check on the association condition every TTI. By considering CRE bias, β, the condition that UE pass and count one check will be: $R_m < \beta R_s$. By considering transition matrix, this is also the condition that M1 move to M2 state, which is the transition probability *Pm(x)*. Sub in (5.8) and (5.9).

$$\begin{aligned} P_M(x) &= P(R_m < \beta R_s) \\ &= P(P_m l_m g_m < \beta P_s l_s g_s) \\ &= P\left(\frac{g_m}{g_s} < \frac{\beta P_s l_s}{P_m l_m}\right) \\ &= 1 - P\left(\frac{g_m}{g_s} > \frac{\beta P_s l_s}{P_m l_m}\right) \end{aligned}$$

If we can solve $P\left(\frac{g_m}{g_s} > \frac{\beta P_s l_s}{P_m l_m}\right)$, the transit probability $P_M(x)$ is also solved. We have known that g_s and g_m are two exponential random variables because they follow Raleigh Distribution. The right side is decided once the location of UE is decided, which can be considered as constant within the same Transition Matrix. Therefore, the question can be transferred to: calculate the probability that the ratio of two random exponential variables X_0 and X_1 is larger than one specific constant, *t*. (Since $\frac{\beta P_s l_s}{P_m l_m}$ is not time dependent and none of the symbol will change during whole MCP, we will consider $\frac{\beta P_s l_s}{P_m l_m}$ as constant *t* during deduction, while g_m and g_s are time dependent to be considered as the two variables X_0 and X_1 during deduction) :

$$Suppose\ t = \frac{\beta P_s l_s}{P_m l_m},\ X_0 = \ g_m,\ X_1 = g_s$$

$$P\left(\frac{X_0}{X_1} > t\right) = P(X_0 > tX_1)$$

$$= \iint\limits_{X_0 > tX_1} f_{X_0,X_1}(x_0, x_1) dA_{x_0,x_1}$$

$$= \int_0^{\infty} \int_{x_1 t}^{\infty} f_{X_0}(x_0) f_{X_1}(x_1) dx_0 dx_1$$

$$= \int_0^{\infty} \left(\int_{x_1 t}^{\infty} f_{X_0}(x_0) dx_0 \right) f_{X_1}(x_1) dx_1$$

$$= \int_0^{\infty} (e^{-tx_1 a})\, a e^{-x_1 a} dx_1$$

$$= \frac{1}{1+t}$$

Hence, we can get from the equation above:

$$P\left(\frac{g_m}{g_s} < t\right) = 1 - P\left(\frac{g_m}{g_s} > t\right)$$

$$= 1 - \frac{1}{1+t}$$

$$= \frac{t}{1+t}$$

$$sub\ in\ t = \frac{\beta P_s l_s}{P_m l_m}$$

$$= \frac{\frac{\beta P_s l_s}{P_m l_m}}{1 + \frac{\beta P_s l_s}{P_m l_m}}$$

$$= \frac{\beta P_s l_s}{P_m l_m + \beta P_s l_s}$$

In conclusion,

$$P_M(x) = \frac{\beta P_s l_s}{P_m l_m + \beta P_s l_s} \tag{5.10}$$

$$P_S(x) = \frac{P_m l_m}{P_m l_m + \beta P_s l_s} \tag{5.11}$$

5.2.3 *Markov-Based Mobility Model*

Conventional mobility model has supposed that UE are moving in strait line during the period of simulation time. However, one cause of unnecessary Handover is just the unpredictable mobility model of UE. As a result, strait line mobility model may not be suitable for this simulation. After understanding how Markov chain process works, we have established a new mobility model based on Markov structure.

Markov Chain is a mathematical system that undergoes transitions from one state to another. It is a random process and generally memoryless – it only relies on the current state and not the whole system. Besides the transition probability P for each state, the parameter needs to be considered in Markov Chain is initial distribution matrix can be derived from users' velocity, direction, or initial state. In our simulation, we need to set up the states first. We have defined two types of person here, one is trespasser and the other is stayer. Trespasser means that they are only trespassing this area and won't stay here. Stayers, however, means they come to this area and will stay here for a while – like people come to work. UE have been generated during last part of the job. After that, each UE will be distributed with a label – being trespasser or stayer, moving on foot or mobile (different moving velocity). Then we need to set up the initial state. For trespasser, we have defined four states to mark their moving direction: N (north), S(south), W(west), E(east). For stayer, we add another state: H(hold), meaning they have reached their office and won't move any more. In other words, H is the final state for stayer.

Therefore, the initial state for trespasser should be like [1, 0, 0, 0], and stayer is like [1,0,0,0,0], which means user is currently moving to north. After that, we should also establish probability matrix P. For stayer, it should be the form as below:

$$\begin{array}{cccccc} & H & N & S & W & E \\ H & 1 & 0 & 0 & 0 & 0 \\ N & 0.1 & 0.3 & 0.2 & 0.3 & 0.1 \\ S & 0.1 & 0.2 & 0.2 & 0.2 & 0.3 \\ W & 0.1 & 0.4 & 0.2 & 0.1 & 0.1 \\ E & 0.1 & 0.1 & 0.3 & 0.2 & 0.3 \end{array}$$

And trespasser should have the form as below:

$$\begin{array}{ccccc} & N & S & E & W \\ N & 0.85 & 0.01 & 0.07 & 0.07 \\ S & 0.01 & 0.85 & 0.07 & 0.07 \\ E & 0.07 & 0.07 & 0.85 & 0.01 \\ W & 0.07 & 0.07 & 0.01 & 0.85 \end{array}$$

This matrix suggests that when one person is walking along specific direction ([1 0 0 0] is the initial probability vector if he move towards north), he may have a high preference to continue this direction, and very low probability to turn back. However, as times passing by, the probability of initial vector will keep shifting according to (5.6). The probability of turning for this user will getting higher till he chooses to turn to a new direction at some time point. The probability vector of this user will be reset to the form [0 0 1 0] if he decides to go east this time. Figure 5.2 shows several users mobility projection by applying this Markov mobility model. Stayer will hold once he entered his office. Trespassers either go out of this area or randomly move by keep changing his direction. For this chapter, we have applied the trespassers model.

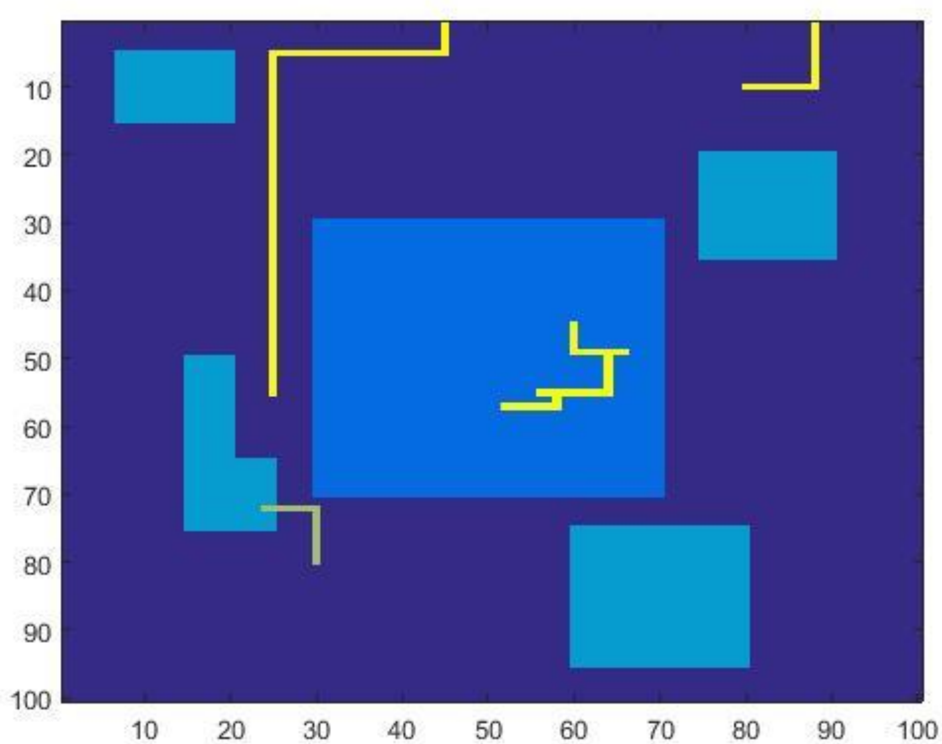

Figure 5-2 Markov Based Mobility Model Simulation

So far, we have defined the MCP model and the transition probability within. The Markov-based mobility model has also been defined. As a result, for any UE within the system, we can use mobility model to get his $P_m l_m$ and $P_s l_s$ first and then calculate the transition probability to construct the transition matrix at any give TTI. At time 0, all UEs are bound to macro cell, and then a small cell is added into the system to establish the HetNets system. With time passing by, UEs around small cell will start to offloading/Handover to small cell. By modelling with MCP, we can calculate the probability distribution of the UE and decide which state it is according to the probability at a specific TTI. UE will continue to move according to his mobility model until 500 TTI is calculated. By calculating the probability of I and I' happens during all TTIs and controlling CRE bias, we can use this model to analyse how CRE bias affect the Handover rate before offloading finishes.

5. 3 Simulation and Analysis

5.3.1 System Model

For this simulation, a two-tier HetNets system is built up with two-BS model, containing single macrocell and single small cell. Two BS are allocated 300 meters away. 500 macro UEs are distributed within the edge of small cell coverage circle to model UE offloading process. Each UE has the moving speed of 3km/h and assigned with MCP trespasser mobility model. Actual simulation parameters are shown below:

Table 5-2 Important simulation parameters for Simulation

Parameters	**Value**
Bandwidth	*1 MHz*
Cell layout	*Two-Tier HetNets*
User Equipment Number	*500*
Transmit power of macro cell	*40W/46dBm*
Transmit power of small cell	*0.25 W/24 dBm*
Noise power	*-174dBm*
TTT	*40 ms*

After time 1, each UE will start moving as his assigned mobility model till TTI reaches 500. Due to the change of location, UE's distance to macro cell and small cell will change accordingly.

We have discussed in early part that CRE has two effects on UE's capacity during Handover phase: (1) mitigating unnecessary Handover rate to boost UE's average capacity and (2) offloading UEs from macro cell small cell, which may hinder UE's capacity by increasing cross-tier interference. Therefore, the effects of these two factors are opposite and increasing CRE bias will inevitably bring the conflict of these two factors. In order to analyse and find the equilibrium point for these two effects, we should try to find out how CRE affects UE capacity when only considering one factor.

$$RSS_m < RSS_s + CRE \qquad (5.12)$$

Firstly, we use conventional user association combining with CRE to offload UE. Whenever macro UE satisfies the (5.13), it will be offloaded to small cell without considering Handover phases (there will be no 0 capacity during any TTI, but also no buffer time for UE, only result is focused). Figure 5.3 shows how total capacity changes with CRE by considering cross-tier interference. It is obvious that CRE has negative effect on capacity because it only focus on solving load balancing issues, and may even increase cross-tier interference. This effect is mild in the beginning and will be more severe as CRE value grows, especially after 9 dB. This means that under low CRE value, less macro UEs will be affected, and those remain in macro cell may ignore cross-tier interference due to high transmission power. However, more UEs are 'forced' to bind with small cell as CRE increases, and even central users of macro cell will be affected. These UEs are more vulnerable to cross-tier interference and will suffer huge QoS loss. After obtaining the total capacity value for different CRE bias value, we have used curve fitting to generate a continuous plot to show the trend and the expression is as follows, where the expression for the curve is:

$$F(x)=-0.004561x^2+0.0002939x+4.739 \qquad (5.13)$$

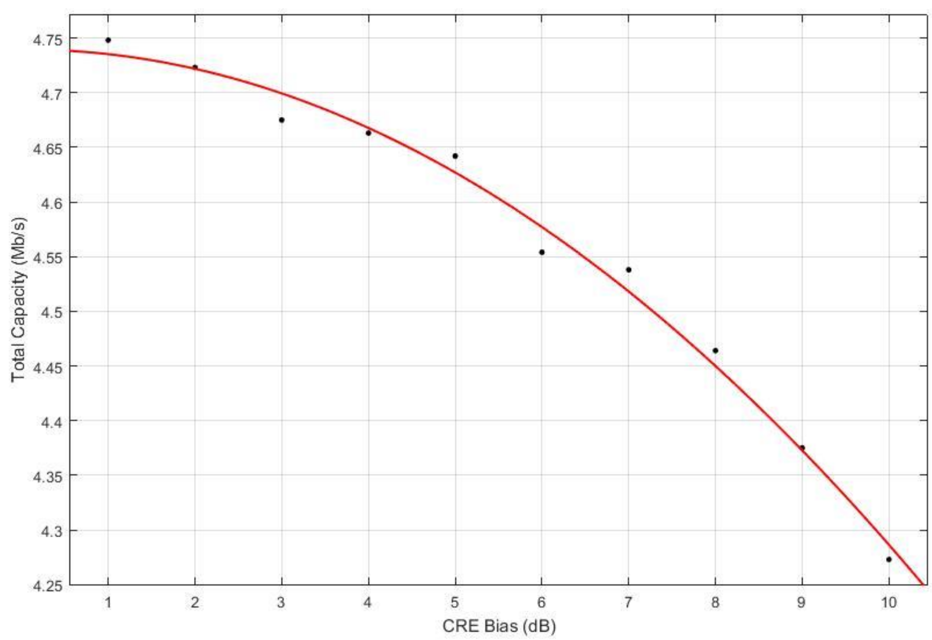

Figure 5-3 Total Capacity vs. CRE for Model 1

Secondly, we use our proposed MCP to model the Handover process and analyse how CRE affects Handover rate for the system. Since CRE's positive effect mainly reflects on the reduction of UE's Handover rate (which means less I or I' states during the process), we keep the UE partition with 1 dB CRE to ignore its effects on offloading even if CRE bias increases. And then combine the capacity of each UE with its Handover rates obtained through MCP under current CRE (we suppose there is no data rate during I and I's states). Figure5.4 displays the result changes with CRE by applying MCP. The effects of CRE is obviously opposite to first model, CRE will take positive effects on capacity due to its ability of controlling Ping-pong handover. However, Figure 5.4 also shows its effect will fade as CRE increases. The phenomena are caused by the property of Ping-pong handover. According to MCP, if CRE is large enough, the probability of unnecessary handovers will be minimized, therefore less Ping-pong handover will occur and the benefit of CRE is diminished. After obtaining the total capacity value for different CRE bias value, we have used curve fitting to generate a continuous plot to show the trend and the expression is as follows:

$$F(x)=-0.006121x^2+0.1452x+3.768 \qquad (5.14)$$

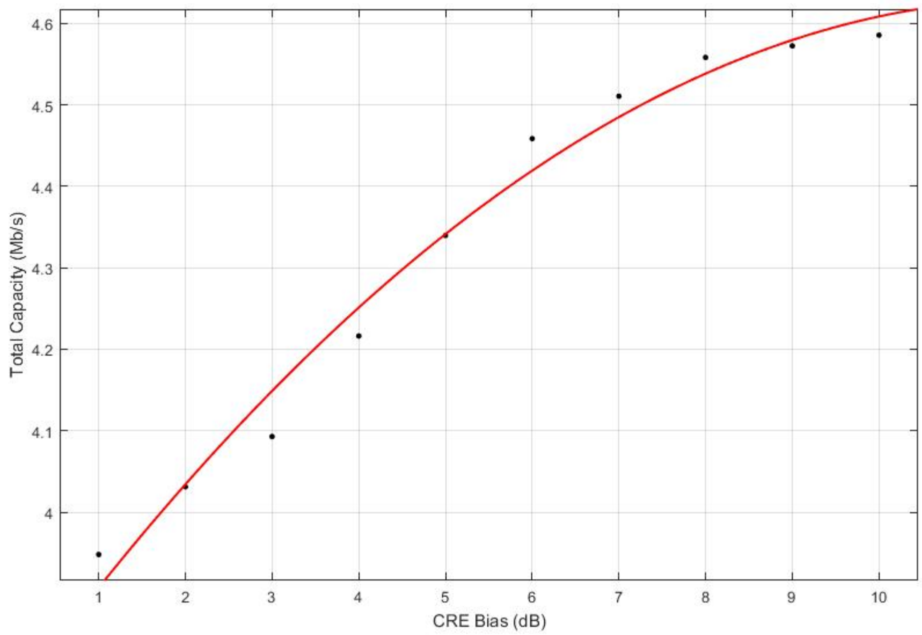

Figure 5-4 Total Capacity vs. CRE for Model 2

Therefore, we may obtain the optimal CRE bias with the help of the two effect-curves. We plot two curves according to their expression on the same graph, and the equilibrium point may be represented by the intersection of the Handover rate effect curve and the cross tier interference curve. Before the equilibrium point, the positive effect of CRE is not fully exploited. And after the point, the negative effects of CRE will getting severer because even core users may be affected. As a result, the point CRE = 7 is the nearest one to equilibrium point, and is presumed to be the optimal CRE bias.

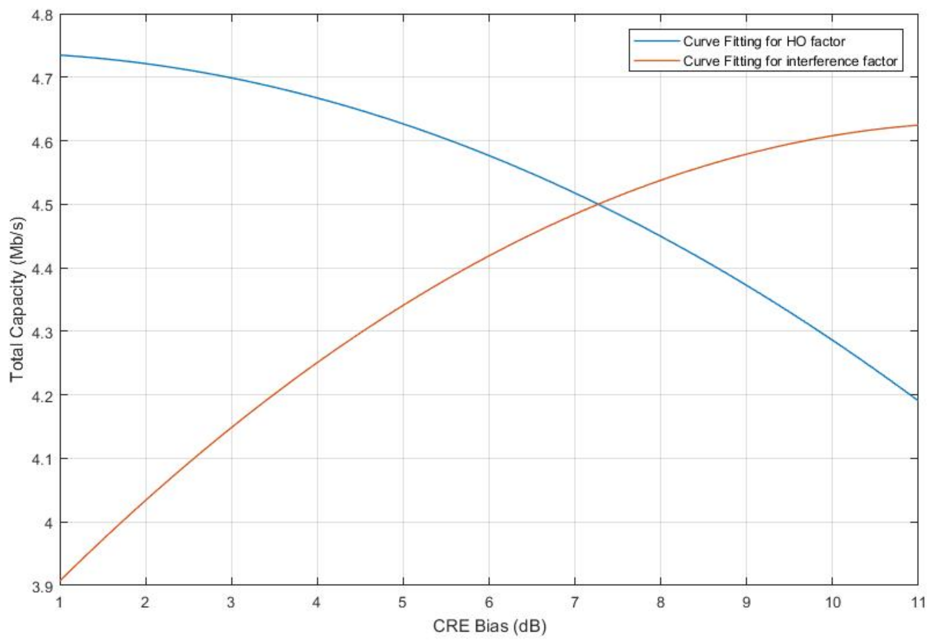

Figure 5-5 Two effect Curves with CRE

We have discussed that the length of TTT will also take the effects on controlling Handover rate. Therefore, we use the established MCP to analyse the effect of TTT. Figure 5.6 shows how handover rate changes with CRE Bias for different TTT values. It can be observed that in four curves, handover rate will decrease with CRE, which follows our prediction by (5.10) and (5.11). It is because that CRE virtually increases coverage of small cells and therefore restrains UEs' handover from small cell to macro cell. With the growth of TTT, the initial handover rate drops dramatically, it suggests that increasing TTT will also benefit controlling handover rate. The ending point for each curve, however, increases slightly as TTT increases. This phenomenon is triggered by the increasing weight of *a* in (5.7), CRE will help to offload UEs from macro cell to small cell. As TTT’s effect on handover drops, CRE may lead a small growth of handover rate for I states in Figure 5.6.

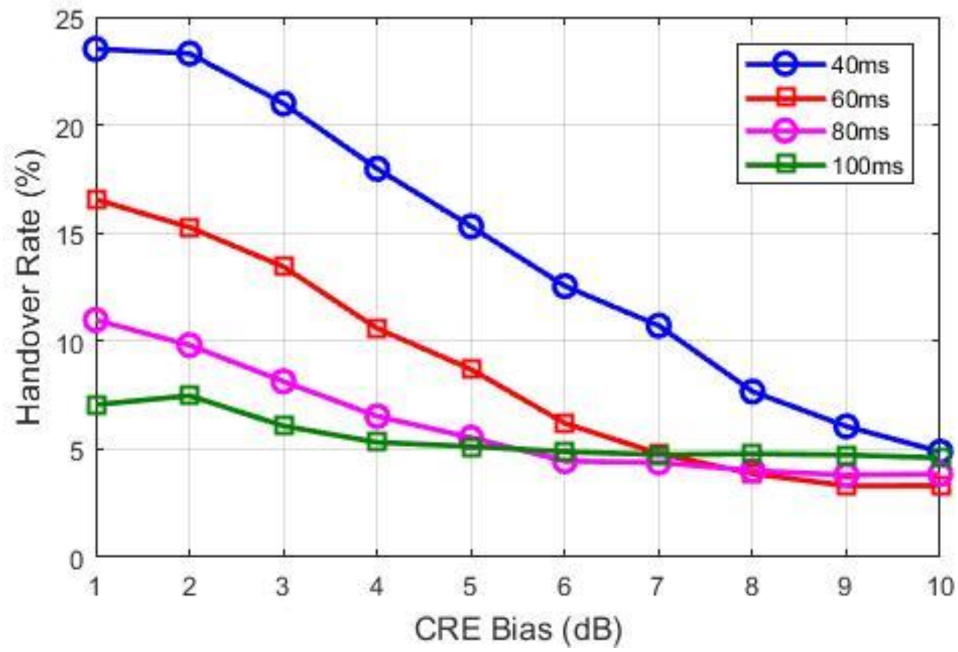

Figure 5-6 Handover Rate vs. CRE under different TTT

We have also discussed that there is another virtual bias HM that is added to help mitigate unnecessary Handover rate. Generally speaking, it can be concluded that both CRE and HM will reduce handover rate as they grow. However, HM's main purpose is to delay the handover process and bound the UEs to their *original* serving cells, which including both macro cell and small cell. CRE, on the other hand, has another function to offload the UEs from macro cell to small cell so that decent HetNets network efficiency can be maintained. The effect of CRE will prefer to bound UEs into small cell. As a result, CRE and HM may take opposite effects on handover control when UEs try to handover from macro cell to small cell. As a result, the modified transition probability after considering both CRE and HM are shown on (5.15) and (5.16).

$$P_m(x) = \frac{\beta P_s l_s}{b P_m l_m + \beta P_s l_s} \quad (5.15)$$

$$P_s(x) = \frac{P_m l_m}{P_m l_m + \beta b P_s l_s} \quad (5.16)$$

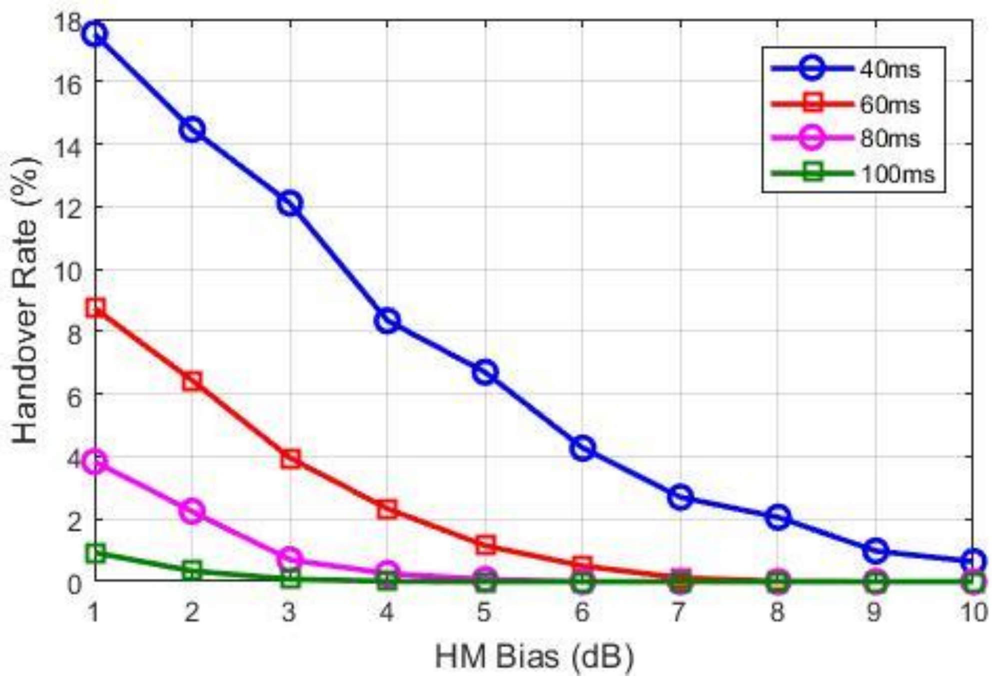

Figure 5-7 Handover Rate vs. HM under different TTT

Figure 5.7 introduces how handover rate changes with HM Bias for different TTT values. It shows that TTT also has a significant effect when HM is applied to control transition probability, which follows the prediction of MCP as well. It can be reflected from two aspects: initial point and reaching-zero bias. When TTT = 40ms, handover rate initial point is up to 18%, after which drops rapidly below 2% when TTT is set to 100ms. Besides that, if HM is set to be extremely high, it is possible that there will be no more handover, UE will not be able to offload to small cell due to high HM and then leave small cell region without handover. As a result, although ping-pong handover is mitigated, it will also hinder our offloading objective for HetNets. The reason causing this side effect, can also be explained from MCP probability formula and physical meanings of two parameters. HM's main purpose is to delay the handover process and bound the UEs to their original serving cells, which including both macro cell and small cell. CRE, on the other hand, has another function to offload the UEs from macro cell to small cell so that decent HetNets network efficiency can be maintained. As a result, CRE and HM may take opposite effects on handover control when UEs try to handover from macro cell to small cell. It also explains why handover rate will not reach 0% no matter what CRE value the network takes. As a result, applying CRE instead of HM to mitigate ping-pong HO may be the suitable scheme for our UE offloading scenario. Furthermore, after discussing how the combination of CRE and HM affect transition probability and obtain (5.15) and (5.16), we can also obtain the optimal combination of CRE and HM.

For CRE only, we have obtained discrete sets result from model 1 and 2, the combined curve suggests that optimal CRE value for this scenario is 7 dB. In order to evaluate the proposed scheme, we calculate each UE’s capacity by considering the UE states in all 500 MCP TTIs, which considers both effects of CRE for this two-tier HetNets scenario. If in I and I’ states, the capacity is 0, if in M states or S states, capacity will be calculated according to respect SINR.

The evaluation is made through CDF as shown Figure5.7. According to the figure, although 5dB has less UEs in low-level tier (0 to 0.03 Mb/s), its distribution increases rapidly in middle-level tier (0.03 to 0.08 Mb/s), it has only a few high-level UEs (0.08 to 0.17 Mb/s).

11dB does have some high-level UEs, but almost 30% of UEs remain in low-level tier. As a result, these two values may not be optimal. 7dB and 9dB behave quite similar though 7dB shows the advantage in high level., it is because that the whole system is in the turning-point state during this period 7dB shows the advantage in high level after 0.06 Mb/s.

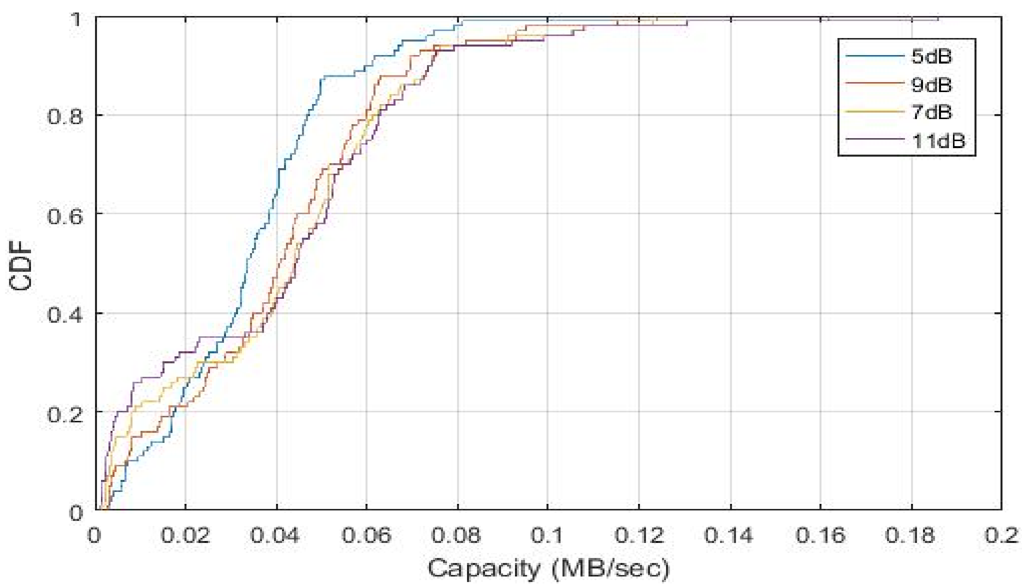

Figure 5-8 CDF of UE capacity for different CRE value

5. 4 Conclusion

Chapter 5 aims to solve ping-pong handover issue during offloading phase within HetNets. Ping-pong Handover can result in communication delay, call dropping, capacity reduction, and this issue may be even more severe in HetNets because of transmission power unbalance. Cell range expansion (CRE), as an important technique of enhanced inter-cell interference coordination (eICIC), can mitigate this issue by adding or reducing the bias on actual received power to enforce user associations; besides, CRE will stabilise UE within specific tier of HetNets and therefore reduce ping-pong handover. However, introducing CRE will also enhance cross-tier interference and decrease QoS, which makes it quite complicated to determine CRE value. This chapter will introduce Markov Chain Process to simulate UE's mobility model and shadow fading randomness when UE is trying to Handover. And then use this MCP system to find the optimal CRE value for different kind of scenarios with Markov Chain Process. Finally, simulation results will show this proposed method's advantage with other fix CRE value method.

Chapter 6. Conclusion and Future Work

In conclusion, the research of this paper can be summarized into three categories. Meanwhile, these three parts will also contribute to solve the challenge of the HetNets. The first part has managed to apply game theory to implement eICIC and FeICIC design so that not only the cross-tier interference is mitigated but also the QoS for all UEs within the system is secured. The second part has applied unsupervised machine learning algorithm to solve the offloading issue of HetNets, which includes offloading macro cell UEs and small cell backhaul load; meanwhile, supervised algorithm has been applied to predict and implement the SON system for later data set. The third part has adopted Markov Chain Process model not only to design a random mobility model but also to model the UE offloading/Handover process from macro cell, so that the optimal CRE during the offloading process can be obtained.

In details, Chapter 3 have managed to solve cross-tier interference issue of HetNets with Almost Blank Subframes. Through muting macro cell in specific ABS, small cell UEs will benefit from it without cross-tier interference. This chapter firstly apply Nash Bargain Solution with proportional fairness to determine the optimal ABS ratio and UE allocation. Which UE are more vulnerable and how ABS affect small cell UEs are also discussed. With the information from ABS, we propose the Power-Layer Based NBS algorithm to realize reducing power ABS. During Rp-ABS, macro cell power is no longer fully muted, we implement the cost of NBS according to power layer and introduce stepped power reduction, so that both the small cell and macro cell UEs may enjoy a system balance. The optimal Rp-ABS ratio and UE allocation for different layer subframe is obtained and evaluated in the end.

Chapter 4 have managed to solve offloading problem of HetNets, which includes (1) UE offloading from higher tier to lower tier, and (2) small cell backhaul traffic offloading. This chapter applies a widely used unsupervised Machine Learning (ML) algorithm, K-means Clustering Algorithm (KCA) to address these two offloading issues. For first issue, we propose a User-Based K-means Algorithm (UBKCA) by involving HetNets background and Enhanced Inter Cell Interference Coordination (eICIC) to decide the optimal Cell Range Expansion (CRE) bias given specific offloading objective. The center user group set is established to reduce computing complexity. Meanwhile, CRE bias and Edge User Factor are introduced to enhance user offloading so that loading balance objective can be achieved. Simulations are then performed to show UBKCA's better performance than KCA; the optimal combination of CRE bias and Edge User Factor are taken based on both accuracy and offloading factor; furthermore, we have implemented a close-loop SON system with KNN and linear classification so that new UE will be automatically assigned to suitable network tier and offloading factor is maintained within a moderate range. In order to solve the backhaul traffic offloading, we have managed to establish a cache system within small cell by applying modified KCA. With the help of the cache system, both small cell users' download speed and request time will be enhanced. KNN is then applied to predict new users' content preference and prove our cache system's suitability. Besides that, we have also proposed a system to predict users' content preference even if the collected data is not complete.

Chapter 3 and chapter 4 manage to solve issue when UEs are in static state. Conversely, chapter 5 aims to solve ping-pong handover issue during offloading phase within HetNets. Ping-pong Handover can result in communication delay, call dropping, capacity reduction, and this issue may be even more severe in HetNets because of transmission power unbalance. Cell range expansion (CRE), as an important technique of enhanced inter-cell interference coordination (eICIC), can mitigate this issue by adding or reducing the bias on actual received power to enforce user associations; besides, CRE will stabilise UE within specific tier of HetNets and therefore reduce ping-pong handover. However, introducing CRE will also enhance cross-tier interference and decrease QoS, which makes it quite complicated to determine CRE value. This chapter will introduce Markov Chain Process to simulate UE's mobility model and shadow fading randomness when UE is trying to Handover. And then use this MCP system to find the optimal CRE value for different kind of scenarios with Markov Chain Process. Finally, simulation results will show this proposed method's advantage with other fix CRE value method.

For the research of chapter 3, Nash Bargain Solution game theory has been applied to obtain the optimal parameter ABS ratio. During the bargain process, it is easy to define the performance of each type of ABS once the structure of the game theory is established. However, the concept of the cost is hard to define because it requires a reasonable explanation and should generate as few negative again as possible. Conventional method applies partial fairness to avoid negative gain, which ignores the cost of players. We have explained and defined the opportunity cost for the game theory instead of partial fairness for FeICIC design. However, we cannot guarantee opportunity cost model is the most suitable one for HetNets. The future work can be focused on finding better cost model, which may lead a higher total utility.

For the research of chapter 4, machine learning algorithms has been applied to solve offloading issue in HetNets. For deciding CRE bias to reach predefined offloading objective, we did not introduce interference factor during training phase of K-means Algorithm. Therefore, the negative effect of CRE could be added during the learning phase so that the SON offloading mechanic will also consider cross-tier interference as a constraint. Moreover, the cache system for small cell backhaul offloading assumes that the small cell storage device is accessible for all UEs within the system. However, the limitation of low transmission power of small cell may cause QoS problem for edge users. We believe that we can also add the geo information as part of UPP through intelligent normalization in future work, so that the clusters will automatically keep away from each other to maintain a range coverage.

For the research of chapter 5, Markov Chain Process has been applied to model the mobility and Handover Process, so that CRE' effect during offloading has been exploited. We have established the Markov-based mobility model to simulate human's behavior when moving. However, the transition probability may not be suitable defined due to the lack of relevant data. In the future, we can find related documents to enhance this mobility model. Moreover, HM is another parameter which may affect the ping-pong handover. We have managed to add it on transition probability but did not make further analyze its effects on handover and its difference between CRE, which may become our future work

References

[1] Chande, V., Sun, H., Vitthaladevuni, P. K., Hou, J., & Mohanty, B. (2010, May). Performance Analysis of 64-qam and mimo in Release 7 WCDMA (HSPA+) Systems. In *Vehicular Technology Conference (VTC 2010-Spring), 2010 IEEE 71st* (pp. 1-5). IEEE.

[2] Iwamura, M., Etemad, K., Fong, M. H., Nory, R., & Love, R. (2010). Carrier aggregation framework in 3GPP LTE-advanced [WiMAX/LTE Update]. *IEEE Communications Magazine*, *48*(8).

[3] Bjerke, B. A. (2011). LTE-advanced and the evolution of LTE deployments. *IEEE Wireless Communications*, *18*(5).

[4] Lopez-Perez, D., Guvenc, I., De la Roche, G., Kountouris, M., Quek, T. Q., & Zhang, J. (2011). Enhanced intercell interference coordination challenges in heterogeneous networks. *IEEE Wireless communications*, *18*(3).

[5] Giuliano, R., Monti, C., & Loreti, P. (2008). WiMAX fractional frequency reuse for rural environments. *IEEE Wireless Communications*, *15*(3).

[6] Okino, K., Nakayama, T., Yamazaki, C., Sato, H., & Kusano, Y. (2011, June). Pico cell range expansion with interference mitigation toward LTE-Advanced heterogeneous networks. In *Communications Workshops (ICC), 2011 IEEE International Conference on* (pp. 1-5). IEEE.

[7] Soret, B., & Pedersen, K. I. (2012, December). Macro transmission power reduction for hetnet co-channel deployments. In *Global Communications Conference (GLOBECOM), 2012 IEEE* (pp. 4126-4130). IEEE.

[8] Cheng, S. M., Lien, S. Y., Chu, F. S., & Chen, K. C. (2011). On exploiting cognitive radio to mitigate interference in macro/femto heterogeneous networks. *IEEE Wireless Communications*, *18*(3).

[9] Oh, J., & Han, Y. (2012, September). Cell selection for range expansion with almost blank subframe in heterogeneous networks. In *Personal Indoor and Mobile Radio Communications (PIMRC), 2012 IEEE 23rd International Symposium on* (pp. 653-657). IEEE.

[10] Lopez-Perez, D., Guvenc, I., & Chu, X. (2012). Mobility management challenges in 3GPP heterogeneous networks. *IEEE Communications Magazine*, *50*(12).

[11] Pauli, V., Naranjo, J. D., & Seidel, E. (2010). Heterogeneous LTE networks and inter-cell interference coordination. *Nomor Research GmBH*, 1-9.

[12] Wang, J., Liu, J., Wang, D., Pang, J., Shen, G., & Chen, J. (2012, May). Load balance based dynamic inter-cell interference coordination for relay enhanced cellular network. In *Vehicular Technology Conference (VTC Spring), 2012 IEEE 75th* (pp. 1-5). IEEE.

[13] González, D., García-Lozano, M., Ruiz, S., & Olmos, J. (2010, September). Static inter-cell interference coordination techniques for LTE networks: A fair performance assessment. In *International Workshop on Multiple Access Communications* (pp. 211-222). Springer, Berlin, Heidelberg.

[14] Ellenbeck, J., Hartmann, C., & Berlemann, L. (2008, June). Decentralized inter-cell interference coordination by autonomous spectral reuse decisions. In *Wireless Conference, 2008. EW 2008. 14th European* (pp. 1-7). IEEE.

[15] Xiao, W., Ratasuk, R., Ghosh, A., Love, R., Sun, Y., & Nory, R. (2006, September). Uplink power control, interference coordination and resource allocation for 3GPP E-UTRA. In *Vehicular Technology Conference, 2006. VTC-2006 Fall. 2006 IEEE 64th* (pp. 1-5). IEEE.

[16] Novlan, T., Andrews, J. G., Sohn, I., Ganti, R. K., & Ghosh, A. (2010, December). Comparison of fractional frequency reuse approaches in the OFDMA cellular downlink. In *Global telecommunications conference (GLOBECOM 2010), 2010 IEEE* (pp. 1-5). IEEE.

[17] Gerlach, C. G., Karla, I., Weber, A., Ewe, L., Bakker, H., Kuehn, E., & Rao, A. (2010). ICIC in DL and UL with network distributed and self-organized resource assignment algorithms in LTE. *Bell Labs Technical Journal*, *15*(3), 43-62.

[18] Ali, M. S. (2015). An overview on interference management in 3GPP LTE-advanced heterogeneous networks. *International Journal of Future Generation Communication and Networking*, *8*(1), 55-68.

[19] Baracca, P., & Aziz, D. (2015, August). Improving eICIC with coordinated beamforming and scheduling in co-channel HetNets. In *Wireless Communication Systems (ISWCS), 2015 International Symposium on* (pp. 581-585). IEEE.

[20] Kamel, M. I., & Elsayed, K. M. (2012, December). Performance evaluation of a coordinated time-domain eICIC framework based on ABSF in heterogeneous LTE-advanced networks. In *Global Communications Conference (GLOBECOM), 2012 IEEE* (pp. 5326-5331). IEEE.

[21] Bedekar, A., & Agrawal, R. (2013, May). Optimal muting and load balancing for eICIC. In *Modeling & Optimization in Mobile, Ad Hoc & Wireless Networks (WiOpt), 2013 11th International Symposium on* (pp. 280-287). IEEE.

[22] Peng, M., Liang, D., Wei, Y., Li, J., & Chen, H. H. (2013). Self-configuration and self-optimization in LTE-advanced heterogeneous networks. *IEEE Communications Magazine*, *51*(5), 36-45.

[23] El-Shaer, H. (2012). Interference management in LTE-Advanced heterogeneous networks using almost blank subframes.

[24] Deb, S., Monogioudis, P., Miernik, J., & Seymour, J. P. (2014). Algorithms for enhanced inter-cell interference coordination (eICIC) in LTE HetNets. *IEEE/ACM transactions on networking*, *22*(1), 137-150.

[25] Crommelinck, M., Feltz, B., & Goujon, P. (Eds.). (2006). *Self-organization and emergence in life sciences*. Springer.

[26] Guidolin, F., Pappalardo, I., Zanella, A., & Zorzi, M. (2014, June). A Markov-based framework for handover optimization in HetNets. In *Ad Hoc Networking Workshop (MED-HOC-NET), 2014 13th Annual Mediterranean* (pp. 134-139). IEEE.

[27] Bellman, R. (1957). A Markovian decision process. *Journal of Mathematics and Mechanics*, 679-684.

[28] Han, Z., Niyato, D., Saad, W., Başar, T., & Hjørungnes, A. (2012). *Game theory in wireless and communication networks: theory, models, and applications*. Cambridge University Press.

[29] Okino, K., Nakayama, T., Yamazaki, C., Sato, H., & Kusano, Y. (2011, June). Pico cell range expansion with interference mitigation toward LTE-Advanced heterogeneous networks. In *Communications Workshops (ICC), 2011 IEEE International Conference on* (pp. 1-5). IEEE.

[30] Krishnan, K. R., & Luss, H. (2011, March). Power selection for maximizing SINR in femtocells for specified SINR in macrocell. In *Wireless Communications and Networking Conference (WCNC), 2011 IEEE* (pp. 563-568). IEEE.

[31] van der Heijden, M. P., Spirito, M., Pelk, M., de Vreede, L. C. N., & Burghartz, J. N. (2004, September). On the optimum biasing and input out-of-band terminations of linear and power efficient class-AB bipolar RF amplifiers. In *Bipolar/BiCMOS Circuits and Technology, 2004. Proceedings of the 2004 Meeting* (pp. 44-47). IEEE.

[32] De La Roche, G., Jaffres-Runser, K., & Gorce, J. M. (2007). On predicting in-building WiFi coverage with a fast discrete approach. *International Journal of Mobile Network Design and Innovation*, *2*(1), 3-12.

[33] Guvenc, I. (2011). Capacity and fairness analysis of heterogeneous networks with range expansion and interference coordination. *IEEE Communications Letters*, *15*(10), 1084-1087.

[34] Crommelinck, M., Feltz, B., & Goujon, P. (Eds.). (2006). *Self-organization and emergence in life sciences*. Springer.

[35] Ramiro, J., & Hamied, K. (Eds.). (2011). *Self-organizing networks: self-planning, self-optimization and self-healing for GSM, UMTS and LTE*. John Wiley & Sons.

[36] López-Pérez, D. (2011). *Practical models and optimization methods for intercell interference coordination in self-organizing cellular networks* (Doctoral dissertation, Ph. D. dissertation, University of Bedforshire, Luton UK).

[37] Engels, A., Reyer, M., Xu, X., Mathar, R., Zhang, J., & Zhuang, H. (2013). Autonomous self-optimization of coverage and capacity in LTE cellular networks. *IEEE Transactions on Vehicular Technology*, *62*(5), 1989-2004.

[38] Sipila, K., Honkasalo, K. C., Laiho-Steffens, J., & Wacker, A. (2000). Estimation of capacity and required transmission power of WCDMA downlink based on a downlink pole equation. In *Vehicular Technology Conference Proceedings, 2000. VTC 2000-Spring Tokyo. 2000 IEEE 51st* (Vol. 2, pp. 1002-1005). IEEE.

[39] Tian, P., Tian, H., Zhu, J., Chen, L., & She, X. (2011). An adaptive bias configuration strategy for range extension in LTE-advanced heterogeneous networks.

[40] Jiang, L., & Lei, M. (2012, September). Resource allocation for eICIC scheme in heterogeneous networks. In *Personal Indoor and Mobile Radio Communications (PIMRC), 2012 IEEE 23rd International Symposium on* (pp. 448-453). IEEE.

[41] Bai, D., Park, C., Lee, J., Nguyen, H., Singh, J., Gupta, A., ... & Kang, I. (2012). LTE-advanced modem design: challenges and perspectives. *IEEE Communications Magazine*, *50*(2).

[42] Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM computing surveys (CSUR)*, *34*(1), 1-47.

[43] Goldman, S., & Zhou, Y. (2000, June). Enhancing supervised learning with unlabeled data. In *ICML* (pp. 327-334).

[44] Chandra, E., & Anuradha, V. P. (2011). A survey on clustering algorithms for data in spatial database management systems. *International Journal of Computer Applications*, *24*(9), 19-26.

[45] Hodge, V., & Austin, J. (2004). A survey of outlier detection methodologies. *Artificial intelligence review*, *22*(2), 85-126.

[46] Gutiérrez Llorente, J. M., Cofiño, A. S., Cano Trueba, R., & Rodríguez, M. Á. (2004). Clustering methods for statistical downscaling in short-range weather forecast.

[47] Ni, S. X., Pearson, N. D., & Poteshman, A. M. (2005). Stock price clustering on option expiration dates. *Journal of Financial Economics*, *78*(1), 49-87.

[48] Kanungo, T., Mount, D. M., Netanyahu, N. S., Piatko, C. D., Silverman, R., & Wu, A. Y. (2002). An efficient k-means clustering algorithm: Analysis and implementation. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (7), 881-892.

[49] Yacoub, M., Badran, F., & Thiria, S. (2001, August). A topological hierarchical clustering: Application to ocean color classification. In *International Conference on Artificial Neural Networks* (pp. 492-499). Springer, Berlin, Heidelberg.

[50] Larose, D. T. (2005). An introduction to data mining. *Traduction et adaptation de Thierry Vallaud*.

[51] Van der Aalst, W. M., Rubin, V., Verbeek, H. M. W., van Dongen, B. F., Kindler, E., & Günther, C. W. (2010). Process mining: a two-step approach to balance between underfitting and overfitting. *Software & Systems Modeling*, *9*(1), 87.

[52] Hawkins, D. M. (2004). The problem of overfitting. Journal of chemical information and computer sciences, 44(1), 1-1

[53] Ulvan, A., Bestak, R., and Ulvan M. (2010). The Study of Handover Procedure in LTE-based Femtocell Network, in *Wireless and Mobile Networking Conference (WMNC)*, (10), 1–6.

[54] Ekiz, N., Salih, T., Kucukoner, S., & Fidanboylu, K. (2005). An overview of handoff techniques in cellular networks. *International journal of information technology*, *2*(3), 132-136.

[55] Aziz, D., & Sigle, R. (2009, April). Improvement of LTE handover performance through interference coordination. In *Vehicular Technology Conference, 2009. VTC Spring 2009. IEEE 69th* (pp. 1-5). IEEE.

[56] Tian, P., Tian, H., Zhu, J., Chen, L., & She, X. (2011). An adaptive bias configuration strategy for range extension in LTE-advanced heterogeneous networks.

[57] López-Pérez, D., Guvenc, I., & Chu, X. (2012, June). Theoretical analysis of handover failure and ping-pong rates for heterogeneous networks. In *Communications (ICC), 2012 IEEE International Conference on* (pp. 6774-6779). IEEE.

[58] Sui, Y., Ren, Z., Sun, W., Svensson, T., & Fertl, P. (2013, December). Performance study of fixed and moving relays for vehicular users with multi-cell handover under co-channel interference. In *Connected Vehicles and Expo (ICCVE), 2013 International Conference on* (pp. 514-520). IEEE.

[59] Karlin, S. (2014). *A first course in stochastic processes*. Academic press.

[60] Saquib, N., Hossain, E., & Kim, D. I. (2013). Fractional frequency reuse for interference management in LTE-advanced hetnets. *IEEE Wireless Communications*, *20*(2), 113-122.

[61] Hamza, A. S., Khalifa, S. S., Hamza, H. S., & Elsayed, K. (2013). A survey on inter-cell interference coordination techniques in OFDMA-based cellular networks. *IEEE Communications Surveys & Tutorials*, *15*(4), 1642-1670.

[62] Molteni, D., & Nicoli, M. (2008, August). Interference mitigation in multicell LTE systems: performance over correlated fading channels. In *Spread Spectrum Techniques and Applications, 2008 IEEE 10th International Symposium on* (pp. 140-144). IEEE.

[63] Bosisio, R., & Spagnolini, U. (2008, March). Interference coordination vs. interference randomization in multicell 3GPP LTE system. In *Wireless Communications and Networking Conference, 2008. WCNC 2008. IEEE* (pp. 824-829). IEEE.

[64] Xia, S., Liang, C., Dai, B., & Hao, P. (2013). *U.S. Patent No. 8,594,017*. Washington, DC: U.S. Patent and Trademark Office.

[65] Molnar, K., Cheng, J. F., & Parkvall, S. (2013). *U.S. Patent No. 8,428,164*. Washington, DC: U.S. Patent and Trademark Office.

[66]Andrews, J. G. (2005). Interference cancellation for cellular systems: a contemporary overview. *IEEE Wireless Communications*, *12*(2), 19-29.

[67] Sawahashi, M., Higuchi, K., Andoh, H., & Adachi, F. (2002). Experiments on pilot symbol-assisted coherent multistage interference canceller for DS-CDMA mobile radio. *IEEE Journal on Selected Areas in Communications*, *20*(2), 433-449.

[68] Varanasi, M. K., & Aazhang, B. (1990). Multistage detection in asynchronous code-division multiple-access communications. *IEEE Transactions on communications*, *38*(4), 509-519.

[69] van der Wijk, F., Janssen, G. M., & Prasad, R. (1995, September). Groupwise successive interference cancellation in a DS/CDMA system. In *Personal, Indoor and Mobile Radio Communications, 1995. PIMRC'95. Wireless: Merging onto the Information Superhighway., Sixth IEEE International Symposium on* (Vol. 2, pp. 742-746). IEEE.

[70] López-Pérez, D., Valcarce, A., De La Roche, G., & Zhang, J. (2009). OFDMA femtocells: A roadmap on interference avoidance. *IEEE Communications Magazine*, *47*(9).

[71] Hicks, J. E., MacKenzie, A. B., Neel, J. A., & Reed, J. H. (2004, December). A game theory perspective on interference avoidance. In *Global Telecommunications Conference, 2004. GLOBECOM'04. IEEE* (Vol. 1, pp. 257-261). IEEE.

[72] Kucera, S., & Lopez-Perez, D. (2016). Inter-cell interference coordination for control channels in LTE heterogeneous networks. *IEEE/ACM Transactions on Networking*, (5), 2872-2884.

[73] Viering, I., Klein, A., Ivrlac, M., Castaneda, M., & Nossek, J. A. (2006, June). On uplink intercell interference in a cellular system. In *Communications, 2006. ICC'06. IEEE International Conference on* (Vol. 5, pp. 2095-2100). IEEE.

[74] Amirteimoori, A., & Tabar, M. M. (2010). Resource allocation and target setting in data envelopment analysis. *Expert Systems with Applications*, *37*(4), 3036-3039.

[75] Navaratnarajah, S., Saeed, A., Dianati, M., & Imran, M. A. (2013). Energy efficiency in heterogeneous wireless access networks. *IEEE Wireless Communications*, *20*(5), 37-43.

[76] Khan, F. (2009). *LTE for 4G mobile broadband: air interface technologies and performance*. Cambridge university press.

[77] Rahman, M., & Yanikomeroglu, H. (2010). Enhancing cell-edge performance: a downlink dynamic interference avoidance scheme with inter-cell coordination. *IEEE Transactions on Wireless Communications*, *9*(4).

[78] Chang, R. Y., Tao, Z., Zhang, J., & Kuo, C. C. J. (2009, June). A Graph Approach to Dynamic Fractional Frequency Reuse (FFR) in Multi-Cell OFDMA Networks. In *ICC* (Vol. 9, pp. 3993-3998).

[79] Porjazoski, M., & Popovski, B. (2010, September). Analysis of intercell interference coordination by fractional frequency reuse in LTE. In *Software, Telecommunications and Computer Networks (SoftCOM), 2010 International Conference on* (pp. 160-164). IEEE.

[80] Jiming, C., Peng, W., & Jie, Z. (2013). Adaptive soft frequency reuse scheme for in-building dense femtocell networks. *China Communications*, *10*(1), 44-55.

[81] Huawei, M. (2005). Soft frequency reuse scheme for UTRAN LTE. *3GPP R1-050507*.

[82] Doppler, K., Wijting, C., & Valkealahti, K. (2009, April). Interference aware scheduling for soft frequency reuse. In *Vehicular Technology Conference, 2009. VTC Spring 2009. IEEE 69th* (pp. 1-5). IEEE.

[83] Liu, J., Wang, D., Pang, J., Wang, J., & Shen, G. (2010, September). Inter-cell interference coordination based on soft frequency reuse for relay enhanced cellular network. In *Personal Indoor and Mobile Radio Communications (PIMRC), 2010 IEEE 21st International Symposium on* (pp. 2304-2308). IEEE.

[84] Chen, L., & Yuan, D. (2009, September). Soft frequency reuse in large networks with irregular cell pattern: How much gain to expect?. In *Personal, Indoor and Mobile Radio Communications, 2009 IEEE 20th International Symposium on*(pp. 1467-1471). IEEE.

[85] Das, S., Viswanathan, H., & Rittenhouse, G. (2003, April). Dynamic load balancing through coordinated scheduling in packet data systems. In *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies* (Vol. 1, pp. 786-796). IEEE.

[86] Bonald, T., Borst, S., & Proutiere, A. (2005, April). Inter-cell scheduling in wireless data networks. In *Wireless Conference 2005-Next Generation Wireless and Mobile Communications and Services (European Wireless), 11th European* (pp. 1-7). VDE.

[87] Koutsimanis, C., & Fodor, G. (2008, May). A dynamic resource allocation scheme for guaranteed bit rate services in OFDMA networks. In *Communications, 2008. ICC'08. IEEE International Conference on* (pp. 2524-2530). IEEE.

[88] Ahmad, I., & Ghafoor, A. (1991). Semi-distributed load balancing for massively parallel multicomputer systems. *IEEE Transactions on Software Engineering*, *17*(10), 987-1004.

[89] Rahman, M., & Yanikomeroglu, H. (2007, November). Multicell downlink OFDM subchannel allocations using dynamic intercell coordination. In *Global Telecommunications Conference, 2007. GLOBECOM'07. IEEE* (pp. 5220-5225). IEEE.

[90] Li, G., & Liu, H. (2006). Downlink radio resource allocation for multi-cell OFDMA system. *IEEE Transactions on wireless communications*, *5*(12).

[91] Necker, M. C. (2011). A novel algorithm for distributed dynamic interference coordination in cellular networks. *PIK-Praxis der Informationsverarbeitung und Kommunikation*, *34*(2), 97-101.

[92] Deb, S., Monogioudis, P., Miernik, J., & Seymour, J. P. (2014). Algorithms for enhanced inter-cell interference coordination (eICIC) in LTE HetNets. *IEEE/ACM transactions on networking*, *22*(1), 137-150.

[93] Amara, M., & Feki, A. (2015, December). Optimized ABS in LTE-advanced heterogeneous networks with adaptive macro cell transmission. In *Globecom Workshops (GC Wkshps), 2015 IEEE* (pp. 1-7). IEEE.

[94] Chen, Y., Fang, X., & Huang, B. (2013, October). Joint ABS power and resource allocations for eICIC in heterogeneous networks. In *IWSDA* (pp. 92-95).

[95] Soret, B., Wang, H., Pedersen, K. I., & Rosa, C. (2013). Multicell cooperation for LTE-advanced heterogeneous network scenarios. *IEEE Wireless Communications*, *20*(1), 27-34.

[96] Malladi, D. P. (2012, May). Heterogeneous networks in 3g and 4g. In *Proc. of IEEE Communication Theory Workshop, Hawaii, USA*.

[97] Wang, X., Wang, C., Cai, R., Huang, S., Wang, C., & Wang, W. (2014, October). Reduced power centralized eICIC for LTE-advanced heterogeneous networks. In *Communications in China (ICCC), 2014 IEEE/CIC International Conference on*(pp. 743-747). IEEE.

[98] Simsek, M., Bennis, M., & Czylwik, A. (2012, December). Dynamic inter-cell interference coordination in HetNets: A reinforcement learning approach. In *Global Communications Conference (GLOBECOM), 2012 IEEE* (pp. 5446-5450). IEEE.

[99] Wang, Y., & Pedersen, K. I. (2011, September). Time and power domain interference management for LTE networks with macro-cells and HeNBs. In *Vehicular Technology Conference (VTC Fall), 2011 IEEE* (pp. 1-6). IEEE.

[100] Gao, L., Tian, H., Tian, P., Zhang, J., & Wang, M. (2013, June). A distributed dynamic ABS ratio setting scheme for macro-femto heterogeneous networks. In *Communications Workshops (ICC), 2013 IEEE International Conference on* (pp. 1221-1225). IEEE.

[101] Yamamoto, K., & Ohtsuki, T. (2014, September). Parameter optimization using local search for CRE and eICIC in heterogeneous network. In *Personal, Indoor, and Mobile Radio Communication (PIMRC), 2014 IEEE 25th Annual International Symposium on* (pp. 1536-1540). IEEE.

[102] Zhou, H., Ji, Y., Wang, X., & Yamada, S. (2017). eICIC configuration algorithm with service scalability in heterogeneous cellular networks. *IEEE/ACM Transactions on Networking (TON)*, *25*(1), 520-535.

[103] Liu, D., Chen, Y., Chai, K. K., & Zhang, T. (2013, October). Performance evaluation of Nash bargaining solution based user association in HetNet. In *Wireless and Mobile Computing, Networking and Communications (WiMob), 2013 IEEE 9th International Conference on* (pp. 571-577). IEEE.

[104] Kamel, M. I., & Elsayed, K. M. (2013, June). ABSF offsetting and optimal resource partitioning for eICIC in LTE-Advanced: Proposal and analysis using a Nash bargaining approach. In *Communications (ICC), 2013 IEEE International Conference on* (pp. 6240-6244). IEEE.

[105] Somasundaram, K. (2013, September). Proportional fairness in lte-advanced heterogeneous networks with eicic. In *Vehicular Technology Conference (VTC Fall), 2013 IEEE 78th* (pp. 1-6). IEEE.

[106] Jiang, L., & Lei, M. (2012, September). Resource allocation for eICIC scheme in heterogeneous networks. In *Personal Indoor and Mobile Radio Communications (PIMRC), 2012 IEEE 23rd International Symposium on* (pp. 448-453). IEEE.

[107] Kudo, T., & Ohtsuki, T. (2013). Cell range expansion using distributed Q-learning in heterogeneous networks. *EURASIP Journal on Wireless Communications and Networking*, *2013*(1), 61.

[108] Al-Rawi, M. (2012, November). A dynamic approach for cell range expansion in interference coordinated LTE-advanced heterogeneous networks. In *Communication Systems (ICCS), 2012 IEEE International Conference on* (pp. 533-537). IEEE.

[109] López-Pérez, D., & Chu, X. (2011, July). Inter-cell interference coordination for expanded region picocells in heterogeneous networks. In *Computer Communications and Networks (ICCCN), 2011 Proceedings of 20th International Conference on* (pp. 1-6). IEEE.

[110] Navaratnarajah, S., Dianati, M., & Imran, M. A. (2015, August). Analysis of energy efficiency on the cell range expansion for cellular-WLAN heterogeneous network. In *Wireless Communications and Mobile Computing Conference (IWCMC), 2015 International* (pp. 514-519). IEEE.

[111] Singh, N. P., & Singh, B. (2007, December). Effects of soft handover margin under various radio propagation parameters in CDMA cellular networks. In *Wireless Communication and Sensor Networks, 2007. WCSN'07. Third International Conference on* (pp. 1-4). IEEE.

[112] Agrawal, R., Bedekar, A., Gupta, R., Kalyanasundaram, S., Kroener, H., & Natarajan, B. (2014, April). Dynamic point selection for LTE-advanced: Algorithms and performance. In *Wireless Communications and Networking Conference (WCNC), 2014 IEEE* (pp. 1392-1397). IEEE.

[113] Kitagawa, K., Komine, T., Yamamoto, T., & Konishi, S. (2012, September). Performance evaluation of handover in LTE-advanced systems with pico cell range expansion. In *Personal Indoor and Mobile Radio Communications (PIMRC), 2012 IEEE 23rd International Symposium on* (pp. 1071-1076). IEEE.

[114] Kuang, Q., Belschner, J., Bleicher, Z., Droste, H., & Speidel, J. (2015). A measurement-based study of handover improvement through range expansion and interference coordination. *Wireless Communications and Mobile Computing*, *15*(14), 1784-1798.

[115] Cicalo, S., Tralli, V., & Perez-Neira, A. I. (2011, May). Centralized vs distributed resource allocation in multi-cell OFDMA systems. In *Vehicular Technology Conference (VTC Spring), 2011 IEEE 73rd* (pp. 1-6). IEEE.

[116] Stolyar, A. L., & Viswanathan, H. (2008, April). Self-organizing dynamic fractional frequency reuse in OFDMA systems. In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE* (pp. 691-699). IEEE.

[117] Ko, S., Seo, H., Kwon, H., & Lee, B. G. (2010, October). Distributed power allocation for efficient inter-cell interference management in multi-cell OFDMA systems. In *Communications (APCC), 2010 16th Asia-Pacific Conference on* (pp. 243-248). IEEE.

[118] Combes, R., Altman, Z., & Altman, E. (2012). Self-organizing relays: Dimensioning, self-optimization, and learning. *IEEE Transactions on Network and Service Management*, *9*(4), 487-500.

[119] Simsek, M., Bennis, M., & Guvenc, I. (2015). Mobility management in HetNets: a learning-based perspective. *EURASIP Journal on Wireless Communications and Networking*, *2015*(1), 26.

[120] Balevi, E., & Gitlin, R. D. (2017, December). Unsupervised machine learning in 5G networks for low latency communications. In *Performance Computing and Communications Conference (IPCCC), 2017 IEEE 36th International* (pp. 1-2). IEEE.

[121] Aprem, A., Murthy, C. R., & Mehta, N. B. (2013). Transmit power control policies for energy harvesting sensors with retransmissions. *IEEE Journal of Selected Topics in Signal Processing*, *7*(5), 895-906.

[122] Simsek, M., Bennis, M., & Güvenç, I. (2015, March). Context-aware mobility management in HetNets: A reinforcement learning approach. In *Wireless Communications and Networking Conference (WCNC), 2015 IEEE* (pp. 1536-1541). IEEE.

[123] Jiang, C., Zhang, H., Ren, Y., Han, Z., Chen, K. C., & Hanzo, L. (2017). Machine learning paradigms for next-generation wireless networks. *IEEE Wireless Communications*, *24*(2), 98-105.

[124] Alnwaimi, G., Vahid, S., & Moessner, K. (2015). Dynamic heterogeneous learning games for opportunistic access in lte-based macro/femtocell deployments. *IEEE Transactions on Wireless Communications*, *14*(4), 2294-2308.

[125] Onireti, O., Zoha, A., Moysen, J., Imran, A., Giupponi, L., Imran, M. A., & Abu-Dayya, A. (2016). A cell outage management framework for dense heterogeneous networks. *IEEE Transactions on Vehicular Technology*, *65*(4), 2097-2113.

[126] Zhang, C., Zhang, T., Zeng, Z., Cuthbert, L., & Xiao, L. (2010, September). Optimal locations of remote radio units in CoMP systems for energy efficiency. In *Vehicular Technology Conference Fall (VTC 2010-Fall), 2010 IEEE 72nd* (pp. 1-5). IEEE.

[127] Elkotby, H. E., Elsayed, K. M., & Ismail, M. H. (2012, November). Shrinking the reuse distance: Spectrally-efficient radio resource management in D2D-enabled cellular networks with interference alignment. In *Wireless Days (WD), 2012 IFIP*(pp. 1-6). IEEE.

[128] Subramanian, A. P., Deshpande, P., Gao, J., & Das, S. R. (2008, April). Drive-by localization of roadside WiFi networks. In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE* (pp. 718-725). IEEE.

[129] Huang, Z. (1998). Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data mining and knowledge discovery*, *2*(3), 283-304.

[130] Nazeer, K. A., & Sebastian, M. P. (2009, July). Improving the Accuracy and Efficiency of the k-means Clustering Algorithm. In *Proceedings of the world congress on engineering* (Vol. 1, pp. 1-3).

[131] Gates, G. (1972). The reduced nearest neighbor rule (Corresp.). *IEEE transactions on information theory*, *18*(3), 431-433.

[132] Hammouda, K. M., & Kamel, M. S. (2002). Phrase-based document similarity based on an index graph model. In *Data Mining, 2002. ICDM 2003. Proceedings. 2002 IEEE International Conference on* (pp. 203-210). IEEE.

[133] Wagstaff, K., Cardie, C., Rogers, S., & Schrödl, S. (2001, June). Constrained k-means clustering with background knowledge. In *ICML* (Vol. 1, pp. 577-584).

[134] Keller, J. M., Gray, M. R., & Givens, J. A. (1985). A fuzzy k-nearest neighbor algorithm. *IEEE transactions on systems, man, and cybernetics*, (4), 580-585.

[135] Cheng, W., & Hüllermeier, E. (2009). Combining instance-based learning and logistic regression for multilabel classification. *Machine Learning*, *76*(2-3), 211-225.

[136] Rahal, I., & Perrizo, W. (2004, March). An optimized approach for KNN text categorization using P-trees. In *Proceedings of the 2004 ACM symposium on Applied computing* (pp. 613-617). ACM.

[137] Mohamad, I. B., & Usman, D. (2013). Standardization and its effects on K-means clustering algorithm. *Research Journal of Applied Sciences, Engineering and Technology*, *6*(17), 3299-3303.

[138] Qualcomm Incorporated (2013), *The 1000x Data Challenge*. Retrieved from http://www.qualcomm.com/1000x /.

[139] Hur, S., Kim, T., Love, D. J., Krogmeier, J. V., Thomas, T. A., & Ghosh, A. (2011, December). Multilevel millimeter wave beamforming for wireless backhaul. In *GLOBECOM Workshops (GC Wkshps), 2011 IEEE* (pp. 253-257). IEEE.

[140]Ge, X., Cheng, H., Guizani, M., & Han, T. (2014). 5G wireless backhaul networks: challenges and research advances. *IEEE Network*, *28*(6), 6-11.

[141] Woo, S., Jeong, E., Park, S., Lee, J., Ihm, S., & Park, K. (2013, June). Comparison of caching strategies in modern cellular backhaul networks. In *Proceeding of the 11th annual international conference on Mobile systems, applications, and services* (pp. 319-332). ACM.

[142] Cache. (n.d.) in *Oxford Dictionaries* Retrieved from https://en.oxforddictionaries.com/definition/cache

[143] Blasco, P., & Gündüz, D. (2014, June). Learning-based optimization of cache content in a small cell base station. In *Communications (ICC), 2014 IEEE International Conference on* (pp. 1897-1903). IEEE.

[144] Cho, K., Lee, M., Park, K., Kwon, T. T., Choi, Y., & Pack, S. (2012, March). Wave: Popularity-based and collaborative in-network caching for content-oriented networks. In *Computer Communications Workshops (INFOCOM WKSHPS), 2012 IEEE Conference on* (pp. 316-321). IEEE.

[145] Candia, J., González, M. C., Wang, P., Schoenharl, T., Madey, G., & Barabási, A. L. (2008). Uncovering individual and collective human dynamics from mobile phone records. *Journal of physics A: mathematical and theoretical*, *41*(22), 224015.

[146] Statista.com,. (2017). *Most popular YouTube channels as of December 2017, ranked by number of subscribers (in millions).* Retrieved from https://www.statista.com/statistics/277758/most-popular-youtube-channels-ranked-by-subscribers/

[147] Wühr, P., Lange, B. P., & Schwarz, S. (2017). Tears or fears? Comparing gender stereotypes about movie preferences to actual preferences. *Frontiers in psychology*, *8*, 428.

[148] Gamerman, D., & Lopes, H. F. (2006). *Markov chain Monte Carlo: stochastic simulation for Bayesian inference*. Chapman and Hall/CRC.

[149] Juanole, G., & Atamna, Y. (1991, December). Dealing with arbitrary time distributions with the Stochastic Timed Petri Net model-Application to queueing systems. In *Proceedings of the Fourth International Workshop on Petri Nets and Performance Models* (pp. 32-41). IEEE.

[150] Kéry, M., & Royle, J. A. (2008). Hierarchical Bayes estimation of species richness and occupancy in spatially replicated surveys. *Journal of Applied Ecology*, *45*(2), 589-598.

[151] Huang, J., Qian, F., Gerber, A., Mao, Z. M., Sen, S., & Spatscheck, O. (2012, June). A close examination of performance and power characteristics of 4G LTE networks. In *Proceedings of the 10th international conference on Mobile systems, applications, and services* (pp. 225-238). ACM.

[152] Lee, Y., Shin, B., Lim, J., & Hong, D. (2010, October). Effects of time-to-trigger parameter on handover performance in SON-based LTE systems. In *Communications (APCC), 2010 16th Asia-Pacific Conference on* (pp. 492-496). IEEE.