WIRELESS COMMUNICATIONS AND MOBILE COMPUTING
Wirel. Commun. Mob. Comput. 2015; **15**:1171–1187
Published online 21 July 2013 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/wcm.2394

RESEARCH ARTICLE

Self-organising cluster-based cooperative load balancing in OFDMA cellular networks†

Lexi Xu*, Yue Chen, Kok Keong Chai, John Schormans and Laurie Cuthbert

School of Electronic Engineering and Computer Science, Queen Mary University of London, London E1 4NS, U.K.

ABSTRACT

Mobility load balancing (MLB) redistributes the traffic load across the networks to improve the spectrum utilisation. This paper proposes a self-organising cluster-based cooperative load balancing scheme to overcome the problems faced by MLB. The proposed scheme is composed of a cell clustering stage and a cooperative traffic shifting stage. In the cell clustering stage, a user-vote model is proposed to address the virtual partner problem. In the cooperative traffic shifting stage, both inter-cluster and intra-cluster cooperations are developed. A relative load response model is designed as the inter-cluster cooperation mechanism to mitigate the aggravating load problem. Within each cluster, a traffic offloading optimisation algorithm is designed to reduce the hot-spot cell's load and also to minimise its partners' average call blocking probability. Simulation results show that the user-vote-assisted clustering algorithm can select two suitable partners to effectively reduce call blocking probability and decrease the number of handover offset adjustments. The relative load response model can address public partner being heavily loaded through cooperation between clusters. The effectiveness of the traffic offloading optimisation algorithm is both mathematically proven and validated by simulation. Results show that the performance of the proposed cluster-based cooperative load balancing scheme outperforms the conventional MLB.

KEYWORDS

cellular networks; traffic shifting; mobility load balancing; call blocking probability

***Correspondence**

Lexi Xu, Networks Research Group, School of Electronic Engineering and Computer Science, Queen Mary University of London, London E1 4NS, U.K.

E-mail: lexi.xu@eecs.qmul.ac.uk; davidlexi@hotmail.com

1. INTRODUCTION

Orthogonal frequency division multiple access (OFDMA) cellular networks can experience random, time-varying and uneven traffic distribution because of high service variety and user mobility [1–4]. To deal with these challenges, various load balancing (LB) schemes have been drawn into attention from academia and industry [5–18]. LB schemes can be generally categorised into channel borrowing schemes and traffic shifting schemes. In channel borrowing schemes, for example, simple borrowing [6] and channel borrowing without locking [7], a hot-spot cell can borrow the idle spectrum from less-loaded neighbouring cells. This type of scheme is more suitable for cellular networks, which pre-allocate different frequency spectrum to neighbouring cells [1,7]. In traffic shifting schemes, a hot-spot cell offloads the traffic to its less-loaded neighbouring cells [8–18]. In OFDMA networks, for example, Long Term Evolution (LTE) and LTE-Advanced, their neighbouring cells share the co-channel spectrum [2,3], and they leave little space for channel borrowing. Hence, traffic shifting schemes are more suitable for OFDMA cellular networks. Mobility LB (MLB) is an effective method employed by 3GPP to distribute cell load evenly among cells or to transfer part of the traffic from hot-spot cells. This is performed by the means of self-optimisation of handover actions [2]. MLB consists of two stages: The hot-spot cell selects less-loaded neighbouring cells as partners, and then, it calculates the required shifting traffic and adjusts cell-specific handover offset (HO_{off}) towards each partner. HO_{off} enlarges the handover region from the hot-spot cell to the partner. Hence, the cell edge users who satisfy the handover condition will be handed over to its partners. Because the traffic shifting direction is from

†Parts of content in this journal paper were published in *IEEE PIMRC2011* [20] and *European Wireless 2012* [21].

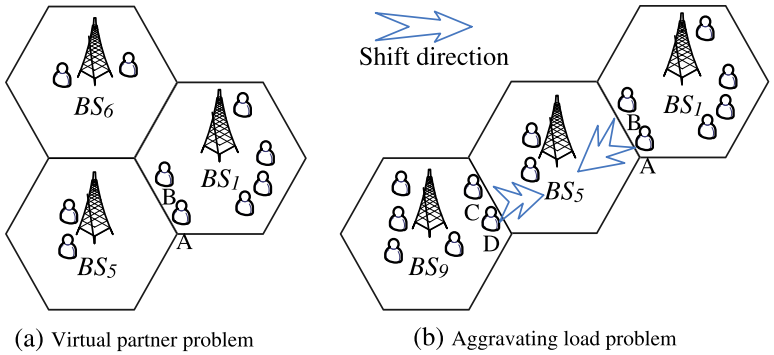


Figure 1. Virtual partner problem and aggravating load problem: (a) virtual partner problem and (b) aggravating load problem.

a hot-spot base station (BS) to each of its partners cells, the MLB schemes in [8–12] calculate the required shifting traffic and adjust the HO_{off}, according to the load difference between the hot-spot BS and its partner. Specifically, in [8], a hot-spot BS selects all lightly loaded neighbouring cells as partners and then calculates the theoretical HO_{off} of each partner on the basis of their load differences. The authors of [9] further researched the precise HO_{off} adjustment, in which a hot-spot BS gradually adjusts HO_{off} with a fixed step-size until the handover users meet the required shifting traffic. In [10], the adaptive step-size-based precise HO_{off} adjustment is designed, to rapidly reach the required shifting traffic. Similarly, the authors of [11] designed a utility function-based adaptive step-size HO_{off} adjustment. The utility function keeps large offset step-size under large load difference between the hot-spot BS and its partners. In [12], a hot-spot BS selects the lowest load neighbouring cell as the partner, and then, the BS gradually adjusts HO_{off} and measures the shifting users until the two cells reach the same load.

As investigated by [19], MLB consumes system signalling load and may result in ping-pong handover and frequent handover. To reduce the number of handovers introduced by frequent traffic shifting, the authors of [13] adopted a lock mechanism to guarantee that a lightly loaded cell can only receive the traffic from only one hot-spot cell at a time. Similarly, the authors of [15] designed an MLB penalty factor to reduce the probability of triggering handover. The authors of [16] researched the urban mobility models of buses and pedestrians. Furthermore, they employed a fuzzy logic controller to adjust HO_{off} and reduce frequent handover in urban simulation scenarios. To mitigate handover failure, the authors of [17,18] designed the load increment estimation-based shifting traffic mechanism, in which a hot-spot cell estimates the partner's load increase introduced by the users it shifted. This mechanism can address shifting a large amount of traffic to the partner.

The first stage of MLB is partner selection. Neighbouring cells with similar load may have different capabilities of serving the shifting users because of users' random location and channel condition. The neighbouring cell's load is a widely used criterion for partner selection; however, this metric cannot differentiate a *virtual partner*. As shown in Figure 1(a), a heavily loaded BS_1 intends to shift traffic out. By applying the load criterion, both BS_5 and BS_6 appear to be possible partners with the same priorities as they have similar load. However, BS_5 is more suitable, whereas BS_6 is a *virtual partner* because BS_6 is far from $User_A$ and $User_B$ and cannot effectively serve them.

When multiple hot-spot BSs shift traffic to one partner, this partner becomes a *public partner* (PP). Without the coordination of hot-spot BSs, their traffic may result in the PP being *heavily loaded*. As shown in Figure 1(b), BS_5 is the PP of both BS_1 and BS_9. The moderate shifting traffic from each BS can result in *heavily loaded* BS_5. In this paper, the phenomenon of heavily loaded PP is called the *aggravating load problem*. To our knowledge, MLB schemes in [8–12,16] did not analyse the coordination of multiple hot-spot cells' traffic shifting towards one PP. The load increment estimation-based shifting traffic mechanism in [17,18] can mitigate the *non-public partner* (NP; receiving traffic from one hot-spot BS) being heavily loaded, although it did not analyse the load increase under *PP* scenario. Because, in distributed control LTE/LTE-Advanced networks, the hot-spot cell cannot estimate the shifting traffic from other hot-spot cells to their PPs, PPs might suffer the aggravating load problem. In [13], a lightly loaded cell can receive traffic from *only one hot-spot cell* at a time. This mechanism avoids the appearance of a heavily loaded PP at the expense of reduced resource utilisation because other hot-spot cells lose the traffic shifting opportunity even though this lightly loaded cell has sufficient idle spectrum to assist other cells.

This paper proposes a self-organising cluster-based cooperative LB scheme that consists of a clustering stage and a cooperative traffic shifting stage. Its aim is to redistribute the traffic among cells and to deal with both the virtual partner problem and the aggravating load problem.

In the clustering stage, after a hot-spot cell identifies itself as a cluster head, the cluster head employs the user-vote model to consider its users' channel condition

received from neighbouring cells. On the basis of both the user-vote model and the neighbouring cell's load, the cluster head selects partners to construct its cluster. Hence, the user-vote-assisted clustering more effectively selects partners and deals with the virtual partner problem, compared with the load-based partner selection.

The cooperative traffic shifting stage researches both inter-cluster cooperation and intra-cluster cooperation. Because multiple cluster heads may select a *PP*, in the inter-cluster cooperation, the PP analyses their traffic shifting requests and then responds with its *relative load* towards each cluster head. Within a cluster, the cluster head's shifting traffic will increase its partners' load and call blocking probabilities. Hence, the cluster head employs the Lagrange multiplier method and the Erlang loss model to optimise its shifting traffic to each partner, to minimise partners' average call blocking probability. In addition, on the basis of the PP's relative load, each cluster head estimates its maximum allowed shifting traffic to the PP, thus addressing the aggravating load problem. In our previous work [20,21], the user-vote model and the basic idea of relative load were introduced. In this paper, our previous work is extended, and we propose a novel traffic offloading optimisation algorithm. The proposed algorithm employs Erlang loss model, Lagrange multiplier method and Karush–Kuhn–Tucker (KKT) conditions to analyse and minimise partners' average call blocking probability in the LB cluster. The cluster head also estimates its maximum allowed shifting traffic to each PP on the basis of PP's relative load. According to the optimization solution and maximum allowed shifting traffic to each PP, the intra-cluster shifting traffic formulas are designed. The traffic offloading optimisation algorithm can minimise partners' average call blocking probability and address the aggravating load problem. Also, in this paper, the relative load response model (RLRM) is analysed comprehensively, including new performance indicator and more reference schemes in simulation, the signalling load and the complexity analysis.

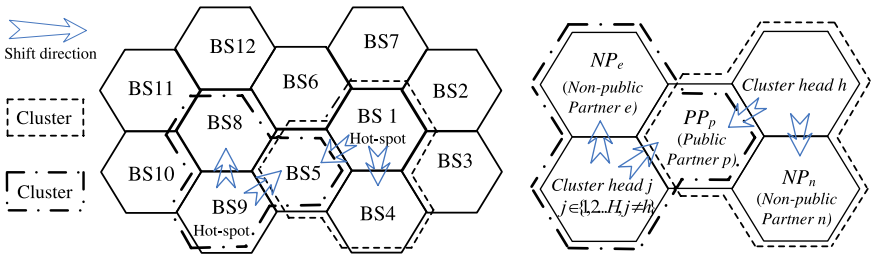
The rest of this paper is organised as follows: Section 2 discusses the system model. Section 3 develops the user-vote-assisted clustering. Section 4 develops the cooperative traffic shifting. Sections 5 and 6 present the simulation analysis and conclusions, respectively.

2. SYSTEM MODEL

An example of the cluster structure is illustrated in Figure 2, where the OFDMA cellular networks suffer an unbalanced load distribution and two clusters are constructed for LB. The hot-spot cell is defined as the *cluster head*, and *partners* are a subset of its neighbouring cells. *Partners* can be classified into two types: A PP receives traffic from multiple hot-spot cells; an NP receives traffic from only one hot-spot cell. Therefore, each cluster is composed of one cluster head and one or more PPs/NPs.

For a more general system model, this paper assumes that the hot-spot BS_h has I neighbouring BSs and BS_h serves K active users. After the clustering stage, there are H cluster heads, which are BS_h and BS_j ($j \in \{1,2\ldots H\}, j \neq h$), requesting to shift their traffic to the *PP* p. In addition, the cluster head BS_h has N NPs indexed with n ($n \in \{1,2\ldots N\}$) and P PPs indexed with p ($p \in \{1,2\ldots P\}$). The definitions and system parameters are as follows:

- M : Total number of subcarriers in each cell.
- M_h : Mean number of subcarriers in use in BS_h, during the load measurement period.
- L : Each BS's actual load. L is defined as the ratio of the number of subcarriers in use to its total subcarriers M, $0\% \leq L \leq 100\%$, for example, the actual load of BS_h $L_h = M_h/M$ [22].
- L_{HL} : Threshold of heavy load/hot-spot. A BS is heavy load/hot-spot when its actual load goes above L_{HL}. (This simulator sets $L_{\mathrm{HL}} = 70\%$. Under 25 physical resource blocks, the call blocking probability of $70\% \times 25$Erlang is 2%, according to Erlang loss model [23,24]).
- BS_i : Neighbouring BS_i. This paper assumes that BS_h has I neighbours indexed with i ($i \in \{1\ldots I\}$).
- U_k : *User* k. This paper assumes BS_h has K active users indexed with k ($k \in \{1\ldots K\}$).
- $SINR^{est}_{k,i}$: Signal to interference noise ratio (SINR) estimation of U_k towards BS_i.
- $SINR_{k,h}$: Serving SINR of U_k in BS_h.
- $V_{k,i}$: *Vote* of U_k towards neighbouring BS_i.
- p : Index of PPs, $p \in \{1\ldots P\}$. PP_p denotes *public partner p*.



Example of two clusters Cluster structure

Figure 2. Example of two clusters and cluster structure.

n : Index of NPs, $n \in \{1...N\}$. NP_n denotes *non-public partner n.*

h, j : Index of cluster heads. PP_p receives traffic from H cluster heads (BS_h and $BS_j (j \in \{1..H\}, j \neq h)$).

L_n : Actual load of NP_n (*non-public partner n*).

L_p : Actual load of PP_p (*public partner p*).

$R_{p,h}$: PP_p's relative load corresponding to the cluster head BS_h.

$\tilde{R}_{p,h}$: PP_p's relative load, after receiving the traffic from the cluster head BS_h.

$\tilde{L}_n$: NP_n's actual load, after receiving traffic from BS_h.

$\tilde{L}_{pars}$: Average load of $BS_{\text{h's}}$ partners, after receiving traffic from BS_h.

$\tilde{B}_n$: NP_n's call blocking probability, after receiving traffic from BS_h.

$\tilde{B}_{p,h}$: PP_p's call blocking probability, after receiving traffic from BS_h.

M_p^{thr} : The receiving traffic threshold of PP_p.

$M_{p,h}^{LB}$: PP_p's LB subcarriers for receiving BS_h's traffic.

3. USER-VOTE-ASSISTED CLUSTERING

This section develops user-vote-assisted clustering. Its objective is to group a small number of neighbouring cells as partners to balance load, and to avoid the virtual partner problem.

3.1. User-vote model

Figure 3(a) shows the cluster head self-discovery mechanism. BS_h discovers itself as a cluster head, when its actual load L_h exceeds the threshold of heavy load L_{HL}, and L_h holds this condition for longer than the critical time T_{crit}. T_{crit} provides hysteresis and helps avoid an incorrect cluster head diagnosis, triggering the cluster construction. The simulator of Section 5 sets $T_{crit} = 5000$ ms [25].

After the cluster head self-discovery, the cluster head employs the user-vote model to construct its cluster. The user-vote model is shown in Figure 3(b). Assume that the cluster head BS_h has I neighbouring BSs indexed with i ($i \in \{1...I\}$) and BS_h has K active users indexed with k ($k \in \{1...K\}$). U_k estimates its SINR from neighbouring BS_i as $SINR_{k,i}^{est}$, and U_k calculates its *vote* of neighbouring BS_i, as $V_{k,i}$. Then, U_k reports two neighbouring BSs with the largest non-zero *vote* $V_{k,i}$ to the cluster head.

SINR estimation

U_k estimates its worst SINR from neighbouring BS_i, on the basis of the reference signal received power (RSRP). In OFDMA networks, the high cell capacity requires the full frequency reuse [2,4]. Hence, all neighbouring BSs are likely to use the co-channel subcarriers of U_k for transmission at the same time, which induces the inter-cell interference. In addition, the precise SINR estimation is difficult because U_k'*s* allocated subcarriers by BS_i are time-varying, on the basis of the channel condition. Therefore, U_k estimates its worst SINR from BS_i by using Equation (1). $SINR_{k,i}^{est}$ reflects the potential data rate after U_k is shifted.

$$SINR_{k,i}^{est} = \frac{RSRP_{k,i}}{RSRP_{k,h} + \sum_{\bar{i}=1,\bar{i}\neq i}^{I} RSRP_{k,\bar{i}}} \quad (1)$$

where $\sum_{\bar{i}=1,\bar{i}\neq i}^{I} RSRP_{k,\bar{i}}$ is from other neighbouring BSs. $RSRP_{k,i}$ and $RSRP_{k,h}$ are from the voting target BS_i and the cluster head BS_h, respectively. In Equation (1), the noise is negligible compared with the interference.

Vote calculation and vote report

On the basis of $SINR_{k,i}^{est}$ and the serving $SINR_{k,h}$ from BS_h, U_k calculates its *vote* as $V_{k,i}$ by using

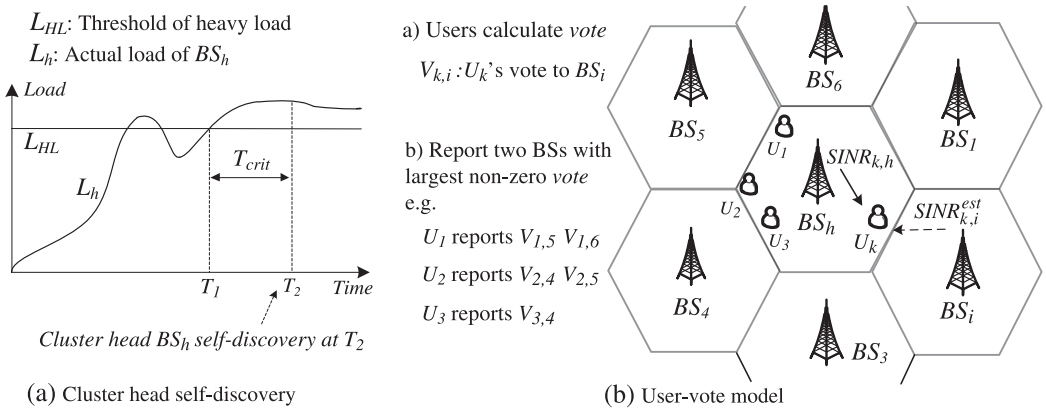


Figure 3. User-vote-assisted clustering diagram: (a) cluster head self-discovery and (b) user-vote model.

Equation (2). $V_{k,i}$ indicates U_k's probability of being offloaded to BS_i, reflecting its satisfaction degree to BS_i

$$V_{k,i} = \begin{cases} 1 & SINR^{est}_{k,i} \geq \dfrac{SINR_{k,h}}{\eta} \\ Q_{step} \times Floor\left(\dfrac{SINR^{est}_{k,i}}{SINR_{k,h}/\eta} \times \dfrac{1}{Q_{step}} + 0.5\right) & SINR^{est}_{k,i} < \dfrac{SINR_{k,h}}{\eta} \end{cases} \quad (2)$$

where $\eta = 4$ to obtain an appropriate threshold to assist U_k to calculate its *vote* to BS_i. It is because $SINR^{est}_{k,i} \geq \frac{SINR_{k,h}}{4}$ can identify cell edge users, and that $\eta = 4$ is a suitable value analysed in Appendix A.

- For the users with $SINR^{est}_{k,i} \geq \frac{SINR_{k,h}}{\eta}$, they are located at the cell edge of BS_h-to-BS_i, and BS_i can serve them with a satisfactory data rate. Hence, they vote for BS_i with full vote $V_{k,i} = 1$.
- For the users with $SINR^{est}_{k,i} < \frac{SINR_{k,h}}{\eta}$, $V_{k,i}$ is based on the ratio of $SINR^{est}_{k,i}$ to $\frac{SINR_{k,h}}{\eta}$.

In Equation (2), $\frac{SINR^{est}_{k,i}}{SINR_{k,h}/\eta}$ is converted to a discrete $V_{k,i}$ value by the quantization step Q_{step} and the *Floor-function*, for example, $V_{k,i} \in \{0, 0.1, 0.2 \ldots 0.9, 1.0\}$ under $Q_{step} = 0.1$.

U_k only reports the *vote* for the two neighbouring BSs with the largest non-zero $V_{k,i}$ to the cluster head. Since in most cases, U_k is near to two neighbouring BSs, and U_k can be shifted to two neighbouring BSs at most. The non-zero constraint avoids the users, which are very near to BS_h, reporting.

3.2. Partner selection

On the basis of the *vote* report of its users, the cluster head calculates the total votes of neighbouring BS_i as $\sum_{k=1}^{K} V_{k,i}$. $\sum_{k=1}^{K} V_{k,i}$ reflects the traffic shifting capability of BS_i, affected by users' channel condition. The higher the value, the more users can shift traffic to BS_i.

The cluster head also considers the actual load, which reflects the idle subcarriers of neighbouring BS_i to serve the shifting users. Therefore, BS_h exchanges the information of actual load with its neighbouring BSs. This process can be implemented over the X2 interface in LTE [26]. The cluster head considers vote and actual load and calculates the selection priority of BS_i as

$$\mathrm{Pr}_i = \frac{\sum_{k=1}^{K} V_{k,i}}{K} + (1 - L_i) \qquad i \in \{1 \ldots I\} \quad (3)$$

where L_i is the actual load of BS_i and K is the total number of active users in the cluster head. The denominator K guarantees that the range of total votes is from 0 to 1, which is the same as the actual load ($0 \leq L_i \leq 1$). Therefore, the factor of actual load and the factor of users' vote have the same weight in Equation (3).

The cluster head also employs two filters to further improve the efficiency of the clustering. The *vote filter* is to avoid selecting a neighbouring BS, which has no user from the cluster head located at its edge, as shown in Equation (4). The *load filter* is to avoid selecting a heavily loaded BS, as shown in Equation (5).

$$\textit{Vote filter:} \quad \max_{k \in \{1 \ldots K\}} V_{k,i} = 1 \qquad i \in \{1 \ldots I\} \quad (4)$$

$$\textit{Load filter:} \quad L_i < L_{HL} \qquad i \in \{1 \ldots I\} \quad (5)$$

where L_{HL} is the threshold of heavy load. In the last step of clustering, the cluster head finds all neighbouring BSs meeting the filters in Equations (4) and (5). Then, the cluster head sorts these neighbouring BSs in descending order, according to their priorities in Equation (3). It continuously selects the highest priority neighbouring BS as cluster's partner in sequence, until the number of partners in the cluster is larger than the maximum cluster size. (Section 5.1 researches the appropriate cluster size via simulation analysis.) Then, the cluster head sends a cluster construction request to the selected neighbouring BSs. The clustering algorithm is finished after their confirmation message.

After the cluster construction, the cluster head shifts traffic to its partners, and this stage is developed in Section 4. After traffic shifting, the cluster head sends the *leave* request to all partners within its cluster. The cluster will be dismissed after partners respond to *leave*.

3.3. Signalling load and complexity

This subsection analyses the signalling load and computational complexity of the user-vote-assisted clustering algorithm. First, users' SINR estimation is purely based on RSRP, which is available for existing resource management functions such as cell selection, and does not require extra measurements. Second, the *vote* report process consumes the signalling load of air interface. To slightly increase the signalling load, the actual SINR ratio in Equation (2) is converted to the discrete *vote* $V_{k,i}$. In addition, each user only reports its *vote* of two neighbouring BSs with the largest non-zero *vote*, rather than reporting all neighbouring BSs' *vote*. Third, the cluster head sends/responds clustering request with partners via cell-to-cell communication. This process consumes the similar

signalling load as the partner request/response process in conventional MLB schemes [8–18].

The complexity of each user calculating the *vote* of all neighbouring BSs is $I \times O(I)$; the complexity of reporting the two largest $V_{k,i}$ neighbouring BSs is $O(2I-3)$. Hence, the complexity of the user-vote model is $K \times I \times O(I) + K \times O(2I-3)$. In the partner selection step, the complexity of priority calculation and the filters of each neighbouring BS are $O(K)$ and $O(K) + O(1)$, respectively. Hence, the complexity of the partner selection is $I \times 2 \times O(K) + I \times O(1)$. Therefore, its overall complexity is $K \times I \times O(I) + K \times O(2I-3) + 2I \times O(K) + I \times O(1)$. Because of the user-vote model, the complexity is higher than load-based partner selection. For example, the complexity of partner selection in [8,9] is $I \times O(1)$, that in [12] is $O(I)$, and that in [13,14] is $I \times O(1) + O(I)$.

4. COOPERATIVE TRAFFIC SHIFTING

After clustering, the cluster head is associated with one or more partner cells. Section 4 presents the cooperative traffic shifting algorithm. Its aim is effectively shifting the cluster head's traffic to deal with the aggravating load problem, as well as minimising the partners' call blocking probabilities. Figure 4 shows its process under the clusters structure of Figure 2. The *PP* analyses traffic shifting requests of multiple cluster heads and replies to each cluster head with its cluster-specific relative load, thus mitigating the aggravating load problem. Meanwhile, the *NP* replies its actual load to the dedicated cluster head. Then, the cluster head employs the traffic offloading optimisation algorithm to calculate the shifting traffic to each partner, to minimise partners' average call blocking probability. On the basis of the required shifting traffic, the cluster head adjusts HO_{off} to offload users. This stage includes inter-cluster coordination and intra-cluster cooperation.

4.1. Inter-cluster cooperation: relative load response model

Relative load response model is the key to inter-cluster cooperation, which assists the PP to coordinate multiple clusters' traffic shifting requests and to address the aggravating load problem. Its basic idea is that the PP analyses its threshold of idle spectrum for receiving traffic. Then, it pre-allocates the idle spectrum to each cluster head's shifting traffic. Finally, the PP calculates its cluster-specific relative load and reports to the corresponding cluster head. On the basis of the relative load, each cluster can shift an appropriate amount of traffic, thus avoiding the PP becoming heavily loaded.

PP's LB spectrum analysis

There are H different cluster heads requests offload traffic to PP_p (PP p). To address heavily loaded PP_p, Equation (6) shows that PP_p's receiving traffic $M_p^{LB} \leq (L_{HL} - L_p) \times M$. Then, PP_p calculates its receiving traffic threshold M_p^{thr} as

$$L_p + \frac{M_p^{LB}}{M} \leq L_{HL} \Rightarrow M_p^{LB} \leq (L_{HL} - L_p) \times M \Rightarrow M_p^{thr} = (L_{HL} - L_p) \times M \quad (6)$$

where L_{HL} is the threshold of heavy load, L_p is the actual load of PP_p and M is the total number of subcarriers in each cell. Equation (6) shows that PP_p's subcarriers for shifted users cannot exceed M_p^{thr} to avoid heavily loaded PP_p. Then, RLRM pre-allocates these M_p^{thr} subcarriers to each cluster head.

These H cluster heads consist of BS_h and BS_j ($j \in \{1, 2 \ldots H\}, j \neq h$). Therefore, PP_{p} pre-allocates these M_p^{thr} subcarriers into two parts: the LB subcarriers for receiving traffic from BS_h

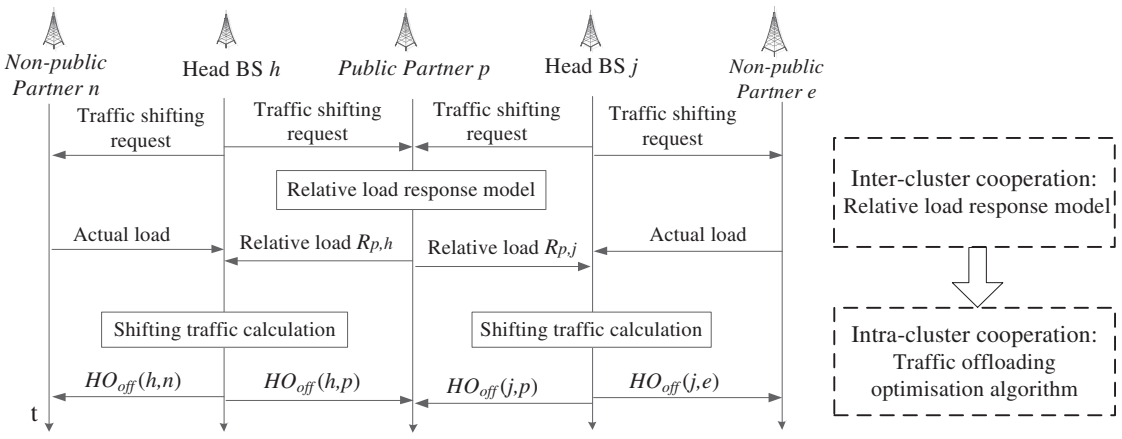


Figure 4. Process of cooperative traffic shifting.

as $M^{LB}_{p,h}$ and the LB subcarriers for receiving traffic from BS_j ($j \in \{1...H\}$, $j \neq h$) as $\sum_{j=1,j\neq h}^{H} M^{LB}_{p,j}$. PP_p pre-allocates more idle subcarriers to a higher-loaded cluster head. Hence, $M^{LB}_{p,h}$ is based on BS_h's actual load L_h, using

$$M^{LB}_{p,h} = M^{thr}_{p} \times \frac{L_h}{L_h + \sum_{\check{j}=1,\check{j}\neq h}^{H} L_{\check{j}}} = \frac{L_h \times M \times (L_{HL} - L_p)}{L_h + \sum_{\check{j}=1,\check{j}\neq h}^{H} L_{\check{j}}} \quad (7)$$

The shifting traffic from BS_h cannot exceed $M^{LB}_{p,h}$. Similarly, PP_p's LB subcarriers for BS_j, $\sum_{j=1,j\neq h}^{H} M^{LB}_{p,j}$ is calculated on the basis of BS_j's actual load L_j, using

$$\sum_{\substack{j=1\\ j\neq h}}^{H} M^{LB}_{p,j} = \sum_{\substack{j=1\\ j\neq h}}^{H} \frac{M^{thr}_{p} \times L_j}{L_h + \sum_{\check{j}=1,\check{j}\neq h}^{H} L_{\check{j}}} = \sum_{\substack{j=1\\ j\neq h}}^{H} \frac{L_j \times M \times (L_{HL} - L_p)}{L_h + \sum_{\check{j}=1,\check{j}\neq h}^{H} L_{\check{j}}} \quad (8)$$

Cluster-specific relative load

From PP_p's actual load $L_p = M_p/M$, M_p subcarriers are used by PP_p itself. Hence, BS_h's shifting users cannot use both M_p and PP_p's LB subcarriers for BS_j's traffic $\sum_{j=1,j\neq h}^{H} M^{LB}_{p,j}$. Therefore, PP_p calculates its cluster-specific relative load towards BS_h as $R_{p,h}$, by using Equation (9a). In Equation (9b), the relative load is converted to a discrete value by the quantization step Q_s and the *Floor-function* [27], for example, $R_{ph} \in \{0, 0.1...0.99, 1.0\}$ under $Q_s = 0.01$. Finally, PP_p informs BS_h with $R_{p,h}$ as the response.

$$R_{p,h} \Leftarrow \left(\sum_{\substack{j=1\\ j\neq h}}^{H} M^{LB}_{p,j}/M \right) + L_p \quad (9a)$$

$$\Rightarrow R_{p,h} = Q_s \times Floor\left\{ \left[\sum_{\substack{j=1\\ j\neq h}}^{H} \frac{L_j (L_{HL} - L_p)}{\left(L_h + \sum_{\check{j}=1,\check{j}\neq h}^{H} L_{\check{j}} \right)} + L_p \right] \frac{1}{Q_s} + 0.5 \right\} \quad (9b)$$

$R_{p,h}$ reflects PP_p's traffic shifting capability towards BS_h. The capability is decided by both PP_p's idle spectrum and all clusters' traffic shifting requests. After the relative load response, each cluster head can estimate its maximum shifting traffic to PP_p and set the shifting traffic constraint, to shift an appropriate amount of traffic. In addition, the relative load of the PP is always higher than its actual load, and hence, the cluster head can shift more traffic to other NPs and less traffic to the PP. Therefore, RLRM assists the PP to coordinate multiple clusters' traffic shifting and address the heavily loaded PP.

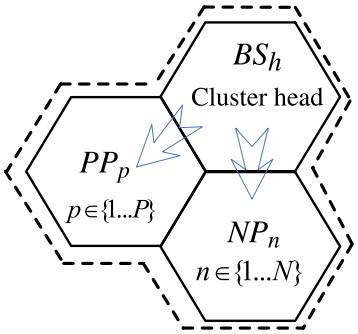


Figure 5. Cluster model of BS_h.

4.2. Intra-cluster cooperation: traffic offloading optimisation algorithm

After the load report stage, different LB schemes have different load reduction objectives for the cluster head BS_h [6–18]. For example, some schemes try to reduce the hot-spot cell's load to the lightly loaded threshold, whereas some other schemes try to reduce the load to its neighbouring cells' average load. To design an LB scheme to meet different load reduction requirements, this paper does not pre-define the load reduction value/threshold. Instead, the proposed scheme assumes that BS_h tries to release $\triangle M_h$ subcarriers, which has different values according to different LB objectives.

Figure 5 shows the cluster model of BS_h introduced in Section 2. This paper assumes that BS_h has N NPs denoted as $NP_n n \in \{1..N\}$ and P PPs denoted as $PP_p p \in \{1..P\}$. Because BS_h tries to offload $\triangle M_h$ traffic to its partners, its load reduction $\Delta L_h = \Delta M_h/M$. This will increase its partners' load and call blocking probability. In this paper, the load and call blocking probability of BS_h's partners are listed as follows:

- Initial load: $L_1 \ldots L_n \ldots L_N$ of NPs; $R_{1,h} \ldots R_{p,h} \ldots R_{P,h}$ of PPs;

- Load after receiving BS_h traffic: $\tilde{L}_1..\tilde{L}_n..\tilde{L}_N$ of NPs; $\tilde{R}_{1,h}..\tilde{R}_{p,h}..\tilde{R}_{P,h}$ of PPs;
- Call blocking probability after receiving BS_h traffic: $\tilde{B}_1...\tilde{B}_n...\tilde{B}_N$ of NPs; $\tilde{R}_{1,h}...\tilde{R}_{p,h}...\tilde{R}_{P,h}$ of PPs.

Therefore, the traffic offloading optimisation algorithm aims at controlling BS_h's shifting traffic to each partner, to minimise these partners' average call blocking probability.

(1) *Optimisation objective: minimise partners' average call blocking probability*

After receiving traffic from BS_h, PP_p's relative load is denoted as $\tilde{R}_{p,h}$, which equals the sum of its relative load $R_{p,h}$ and BS_h's shifting traffic; after traffic shifting, NP_n's actual load is denoted as $\tilde{L}_n$, which equals the sum of its actual load L_n and BS_h's shifting traffic. Therefore, under the cluster head's load reduction $\triangle L_h$, all partners' total load is expressed as

$$\begin{aligned}\tilde{L}_{pars}^{all} &= \Delta L_h + \sum_{n=1}^{N} L_n + \sum_{p=1}^{P} R_{p,h} \\ &= \sum_{n=1}^{N} \tilde{L}_n + \sum_{p=1}^{P} \tilde{R}_{p,h}\end{aligned} \tag{10}$$

The Erlang loss model is widely used to evaluate the grade of service in wireless networks [23,24]. After receiving traffic from BS_h, the call blocking probability of NP_n and PP_p are calculated on the basis of the Erlang loss model, as $\tilde{B}_n$ in Equation (11) and $\tilde{B}_{p,h}$ in Equation (12), respectively.

$$\tilde{B}_n = \frac{\left(\tilde{L}_n \times M\right)^M / M!}{\sum_{k=0}^{M} \left(\tilde{L}_n \times M\right)^k / k!} \quad n \in \{1....N\} \tag{11}$$

$$\tilde{B}_{p,h} = \frac{\left(\tilde{R}_{p,h} \times M\right)^M / M!}{\sum_{k=0}^{M} \left(\tilde{R}_{p,h} \times M\right)^k / k!} \quad p \in \{1....P\} \tag{12}$$

Under BS_h'*s* load reduction $\triangle L_h$, the optimisation objective of minimising its partners' average call blocking probability $\tilde{B}_{pars}$ is formulated as Equations (13)–(16).

$$\underset{\tilde{L}_n, \tilde{R}_{p,h}}{MIN} \tilde{B}_{pars} = \underset{\tilde{L}_n, \tilde{R}_{p,h}}{MIN} \frac{\sum_{n=1}^{N} \tilde{B}_n \tilde{L}_n + \sum_{p=1}^{P} \tilde{B}_{p,h} \tilde{R}_{p,h}}{\tilde{L}_{pars}^{all}} \tag{13}$$

$$\text{Subject to} \quad \tilde{L}_{pars}^{all} - \sum_{n=1}^{N} \tilde{L}_n - \sum_{p=1}^{P} \tilde{R}_{p,h} = 0 \tag{14}$$

$$\tilde{L}_n > L_n \Rightarrow \tilde{L}_n - L_n > 0 \; n \in \{1...N\} \tag{15}$$

$$\tilde{R}_{p,h} > R_{p,h} \Rightarrow \tilde{R}_{p,h} - R_{p,h} > 0 \; p \in \{1...P\} \tag{16}$$

The total load constraint of Equation (14) is derived from Equation (10). Because each NP receives traffic from the cluster head, this will increase the *actual load* of each *NP*, and this constraint is depicted as Equation (15). Similarly, the shifting traffic from the cluster head will increase the *relative load* of each *PP*, and this constraint is depicted as Equation (16).

(2) *Optimisation method*

To minimise $\tilde{B}_{pars}$, this paper uses the *Lagrange multiplier method* and *KKT* conditions [28]. The *Lagrange multiplier* λ is introduced for the constraint of Equation (14). In addition, the *Lagrange multiplier vectors* $\vec{\omega} = \{\omega_1, \omega_2...\omega_N\}$ and $\vec{\mu} = \{\mu_1, \mu_2...\mu_p\}$ are introduced for the constraints of Equations (15) and (16), respectively.

(a) First, the objective formulated in Equations (13)–(16) is defined as the *Lagrangian function*

$$\begin{aligned}&F\left(\tilde{L}_n, \tilde{R}_{p,h}\right) \\ &= \frac{\sum_{n=1}^{N} \tilde{B}_n \tilde{L}_n + \sum_{p=1}^{P} \tilde{B}_{p,h} \tilde{R}_{p,h}}{\tilde{L}_{pars}^{all}} \\ &\quad - \lambda \left(\tilde{L}_{pars}^{all} - \sum_{n=1}^{N} \tilde{L}_n - \sum_{p=1}^{P} \tilde{R}_{p,h} \right) \\ &\quad - \sum_{n=1}^{N} \omega_n \times \left(\tilde{L}_n - L_n\right) \\ &\quad - \sum_{p=1}^{P} \mu_p \times \left(\tilde{R}_{p,h} - R_{p,h}\right)\end{aligned} \tag{17}$$

where $\tilde{B}_n$ and $\tilde{B}_{p,h}$ are the functions of variable $\tilde{L}_n$ and $\tilde{R}_{p,h}$, respectively, as shown in Equations (11) and (12).

According to the *KKT* conditions, for $n \in \{1, 2...N\}$, there is $\omega_n \times \left(\tilde{L}_n - L_n\right) = 0$. Meanwhile, Equation (15) shows $\tilde{L}_n - L_n > 0$. Therefore, the *Lagrange multiplier* $\omega_n = 0$ when $n = 1, 2...N$.

Similarly, the *KKT* conditions require $\mu_p \times \left(\tilde{R}_{p,h} - R_{p,h}\right) = 0$, and Equation (16) has the constraint $\tilde{R}_{p,h} - R_{p,h} > 0$. Therefore, the *Lagrange multiplier* $\mu_p = 0$ when $p = 1, 2...P$.

For the *Lagrange multiplier* λ, Equation (14) shows $\tilde{L}_{pars}^{all} - \sum_{n=1}^{N} \tilde{L}_n - \sum_{p=1}^{P} \tilde{R}_{p,h} = 0$. If λ is zero, these multipliers will lose their impact on the constraints. Therefore, $\lambda \neq 0$.

(b) Second, the partial derivative $\frac{\partial F}{\partial \tilde{L}_n} n \in \{1 \ldots N\}$ and $\frac{\partial F}{\partial \tilde{R}_{p,h}} p \in \{1 \ldots P\}$ are given by Equations (18) and (19).

$$\begin{aligned}\frac{\partial F}{\partial \tilde{L}_n} &= \frac{\partial\left(\tilde{B}_n \times \tilde{L}_n\right)}{\partial \tilde{L}_n}\frac{1}{\tilde{L}_{pars}^{all}} \\ &\quad - \lambda\frac{\partial\left(\tilde{L}_{pars}^{all} - \tilde{L}_n - \sum_{\substack{\tilde{n}=1 \\ \tilde{n}\neq n}}^{N}\tilde{L}_{\tilde{n}} - \sum_{\tilde{p}=1}^{P}\tilde{R}_{\tilde{p},h}\right)}{\partial \tilde{L}_n} \\ &\quad - 0 - 0 = \frac{(M+1)\tilde{B}_n}{\tilde{L}_{pars}^{all}} \\ &\quad - \frac{\sum_{k=0}^{M}\left(\tilde{L}_n \times M\right)^k \times k/k!}{\sum_{k=0}^{M}\left(\tilde{L}_n \times M\right)^k/k!}\frac{\tilde{B}_n}{\tilde{L}_{pars}^{all}} + \lambda\end{aligned} \tag{18}$$

$$\begin{aligned}\frac{\partial F}{\partial \tilde{R}_{p,h}} &= \frac{(M+1)\tilde{B}_{p,h}}{\tilde{L}_{pars}^{all}} \\ &\quad - \frac{\sum_{k=0}^{M}\left(\tilde{R}_{p,h} \times M\right)^k \times k/k!}{\sum_{k=0}^{M}\left(\tilde{R}_{p,h} \times M\right)^k/k!}\frac{\tilde{B}_{p,h}}{\tilde{L}_{pars}^{all}} + \lambda\end{aligned} \tag{19}$$

where $\tilde{B}_n$ function and $\tilde{B}_{p,h}$ function refer to Equations (11) and (12), respectively. Hence, Equation (20) is constructed to obtain the solution of $\frac{\partial F}{\partial \tilde{L}_n} n \in \{1 \ldots N\}$ and $\frac{\partial F}{\partial \tilde{R}_{p,h}} p \in \{1 \ldots P\}$.

$$\begin{cases} \frac{\partial F}{\partial \tilde{L}_1} = \frac{(M+1)\tilde{B}_1}{\tilde{L}_{pars}^{all}} - \frac{\sum_{k=0}^{M}\frac{\left(\tilde{L}_1 \times M\right)^k \times k}{k!}}{\sum_{k=0}^{M}\left(\tilde{L}_1 \times M\right)^k/k!}\frac{\tilde{B}_1}{\tilde{L}_{pars}^{all}} + \lambda = 0 \\ \quad \ldots\ldots\ldots\ldots\ldots\ldots \\ \frac{\partial F}{\partial \tilde{L}_n} = \frac{(M+1)\tilde{B}_n}{\tilde{L}_{pars}^{all}} - \frac{\sum_{k=0}^{M}\frac{\left(\tilde{L}_n \times M\right)^k \times k}{k!}}{\sum_{k=0}^{M}\left(\tilde{L}_n \times M\right)^k/k!}\frac{\tilde{B}_n}{\tilde{L}_{pars}^{all}} + \lambda = 0 \\ \frac{\partial F}{\partial \tilde{R}_{1,h}} = \frac{(M+1)\tilde{B}_{1,h}}{\tilde{L}_{pars}^{all}} - \frac{\sum_{k=0}^{M}\frac{\left(\tilde{R}_{1,h} \times M\right)^k \times k}{k!}}{\sum_{k=0}^{M}\left(\tilde{R}_{1,h} \times M\right)^k/k!}\frac{\tilde{B}_{1,h}}{\tilde{L}_{pars}^{all}} + \lambda = 0 \\ \quad \ldots\ldots\ldots\ldots\ldots\ldots \\ \frac{\partial F}{\partial \tilde{R}_{p,h}} = \frac{(M+1)\tilde{B}_{p,h}}{\tilde{L}_{pars}^{all}} - \frac{\sum_{k=0}^{M}\frac{\left(\tilde{R}_{p,h} \times M\right)^k \times k}{k!}}{\sum_{k=0}^{M}\left(\tilde{R}_{p,h} \times M\right)^k/k!}\frac{\tilde{B}_{p,h}}{\tilde{L}_{pars}^{all}} + \lambda = 0 \end{cases}$$

$$\Rightarrow \begin{cases} \lambda = \frac{\left(\tilde{L}_1 \times M\right)^M}{M!} \times \frac{\sum_{k=0}^{M}\left(\tilde{L}_1 \times M\right)^k (k-M-1)}{k!} \Big/ \tilde{L}_{pars}^{all}\left[\sum_{k=0}^{M}\left(\tilde{L}_1 \times M\right)^k/k!\right]^2 \\ \quad \ldots\ldots\ldots\ldots\ldots\ldots \\ \lambda = \frac{\left(\tilde{L}_n \times M\right)^M}{M!} \times \frac{\sum_{k=0}^{M}\left(\tilde{L}_n \times M\right)^k (k-M-1)}{k!} \Big/ \tilde{L}_{pars}^{all}\left[\sum_{k=0}^{M}\left(\tilde{L}_n \times M\right)^k/k!\right]^2 \\ \lambda = \frac{\left(\tilde{R}_{1,h} \times M\right)^M}{M!} \times \frac{\sum_{k=0}^{M}\left(\tilde{R}_{1,h} \times M\right)^k (k-M-1)}{k!} \Big/ \tilde{L}_{pars}^{all}\left[\sum_{k=0}^{M}\left(\tilde{R}_{1,h} \times M\right)^k/k!\right]^2 \\ \quad \ldots\ldots\ldots\ldots\ldots\ldots \\ \lambda = \frac{\left(\tilde{R}_{p,h} \times M\right)^M}{M!} \times \frac{\sum_{k=0}^{M}\left(\tilde{R}_{p,h} \times M\right)^k (k-M-1)}{k!} \Big/ \tilde{L}_{pars}^{all}\left[\sum_{k=0}^{M}\left(\tilde{R}_{p,h} \times M\right)^k/k!\right]^2 \end{cases} \tag{20}$$

(c) Third, after solving the previous Equation (20), λ can be obtained as Equation (21), and $\tilde{L}_n$ and $\tilde{R}_{p,h}$ can be obtained as Equation (22).

$$\begin{aligned}\lambda &= \frac{\left(\frac{\sum_{n=1}^{N}\tilde{L}_n + \sum_{p=1}^{P}\tilde{R}_{p,h}}{N+P} \times M\right)^M \Big/ M!}{\left(\sum_{n=1}^{N}\tilde{L}_n + \sum_{p=1}^{P}\tilde{R}_{p,h}\right)} \\ &\quad \times \frac{\sum_{k=0}^{M}\frac{\left(\frac{\sum_{n=1}^{N}\tilde{L}_n + \sum_{p=1}^{P}\tilde{R}_{p,h}}{N+P} \times M\right)^k \times [k-(M+1)]}{k!}}{\left[\sum_{k=0}^{M}\left(\frac{\sum_{n=1}^{N}\tilde{L}_n + \sum_{p=1}^{P}\tilde{R}_{p,h}}{N+P} \times M\right)^k \Big/ k!\right]^2}\end{aligned} \tag{21}$$

$$\begin{cases} \tilde{L}_n = \frac{\sum_{\check{n}=1}^{N}\tilde{L}_{\check{n}} + \sum_{\check{p}=1}^{P}\tilde{R}_{\check{p},h}}{N+P} & n \in \{1 \ldots N\} \\ \tilde{R}_{p,h} = \frac{\sum_{\check{n}=1}^{N}\tilde{L}_{\check{n}} + \sum_{\check{p}=1}^{P}\tilde{R}_{\check{p},h}}{N+P} & p \in \{1 \ldots P\} \end{cases} \tag{22}$$

The value of $\frac{\sum_{n=1}^{N}\tilde{L}_n + \sum_{p=1}^{P}\tilde{R}_{p,h}}{N+P}$ is equal to the average load of BS_h's partners after receiving traffic. This paper defines $\tilde{L}_{pars} = \frac{\sum_{n=1}^{N}\tilde{L}_n + \sum_{p=1}^{P}\tilde{R}_{p,h}}{N+P}$.

(d) Solution of minimising partners' average call blocking probability

According to the theoretical analysis from Equation (13) to Equation (22), Equation (22) is the solution of the optimisation objective of minimising partners' average call blocking probability, which is presented in Equations (13)–(16). Equation (22) means that each PP's relative load and NP's actual load reach the same load. Namely, $\tilde{L}_1 = \ldots = \tilde{L}_n = \ldots = \tilde{R}_{p,h} = \ldots = \tilde{R}_{P,h} = \tilde{L}_{pars}$.

From the previous analysis, the partners' average call blocking probability is minimised when the cluster head shifts its traffic until

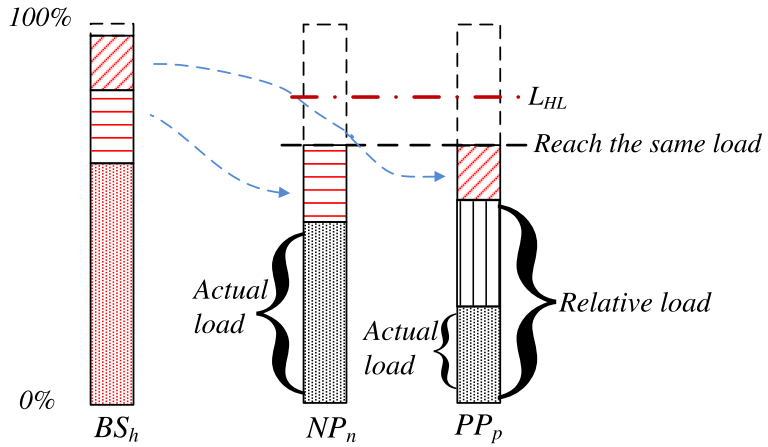


Figure 6. Illustration of the solution of minimising partners' average call blocking probability.

each PP's relative load and each NP's actual load become equal. This is illustrated in Figure 6.

Furthermore, under the cluster head BS_h's shifting load $\triangle L_h$, its partners' theoretical minimal call blocking probability $\tilde{B}_{pars}^{\min}$ is

$$\tilde{B}_{pars} = \frac{\sum_{n=1}^{N} \tilde{L}_n \tilde{B}_n + \sum_{p=1}^{P} \tilde{R}_{p,h} \tilde{B}_{p,h}}{\tilde{L}_{pars}^{all}}$$

$$\tilde{L}_n, \tilde{R}_{p,h} = \frac{\triangle L_h + \sum_{n=1}^{N} L_n + \sum_{p=1}^{P} R_{p,h}}{N+P} \Rightarrow$$

$$\tilde{B}_{pars}^{\min} = \frac{\left[\frac{\triangle L_h + \sum_{n=1}^{N} L_n + \sum_{p=1}^{P} R_{p,h}}{N+P} M\right]^M \Big/ M}{\sum_{k=0}^{M} \left[\frac{\triangle L_h + \sum_{n=1}^{N} L_n + \sum_{p=1}^{P} R_{p,h}}{N+P} M\right]^k \Big/ k!} \tag{23}$$

(3) *Intra-cluster shifting traffic calculation*

On the basis of the solution of minimising partners' average call blocking probability, this work designs the shifting traffic calculation formula. After receiving the traffic of ΔM_h ($\Delta M_h = \Delta L_h \times M$), the average load of BS_h's partners $\tilde{L}_{pars}$ is

$$\begin{aligned} \tilde{L}_{pars} &= \frac{\sum_{n=1}^{N} \tilde{L}_n + \sum_{p=1}^{P} \tilde{R}_{p,h}}{N+P} \\ &= \frac{\frac{\Delta M_h}{M} + \sum_{n=1}^{N} L_n + \sum_{p=1}^{P} R_{p,h}}{N+P} \\ &= \frac{\Delta L_h + \sum_{p=1}^{N} L_p + \sum_{p=1}^{P} R_{p,h}}{N+P} \end{aligned} \tag{24}$$

where L_n is NP_n's actual load before traffic shifting and $R_{p,h}$ is PP_p's relative load towards BS_h before traffic shifting.

(a) Shifting traffic to PP_p

To save the signalling load of cell-to-cell communication [2,26], PP_p does not inform BS_h with $M_{p,h}^{LB}$ $\left(M_{p,h}^{LB}\right.$ is PP_p's LB subcarriers for BS_h as discussed in Section 4.1$\left.\right)$. Hence, BS_h estimates $M_{p,h}^{LB}$ according to the relative load $R_{p,h}$. Formula (9) shows that PP_p allocates $R_{p,h} \times M$ subcarriers to both cluster head BS_j's shifting traffic and PP_p's serving users. Meanwhile, PP_p's actual load cannot exceed the heavily loaded threshold L_{HL}. Hence, BS_h estimates $M_{p,h}^{LB}$ as

$$M_{p,h}^{LB} \approx L_{HL} \times M - R_{p,h} \times M \tag{25}$$

This paper defines the shifting traffic from BS_h to PP_p as $\triangle M_{h,p}$. $\triangle M_{h,p}$ cannot exceed $M_{p,h}^{LB}$ to avoid PP_p being heavily loaded, as shown in Equation (27). On the basis of the solution of minimising partners' average call blocking probability discussed previously, PP_p should receive BS_h's traffic until its relative load $R_{p,h}$ reaches $\tilde{L}_{pars}$. Hence, BS_h uses Equations (26) and (27) to calculate $\triangle M_{h,p}$.

$$\Delta M_{h,p} = \left(\tilde{L}_{pars} - R_{p,h}\right) \times M \quad p \in \{1 \ldots P\} \tag{26}$$

$$\text{Subject to } \Delta M_{h,p} \leq M_{p,h}^{LB} \approx (L_{HL} - R_{p,h}) \times M \tag{27}$$

(b) Shifting traffic to NP_n

To reach $\tilde{L}_{pars}$, the shifting traffic from BS_h to NP_n, $\triangle M_{h,n}$ is calculated using Equations (28) and (29). The constraint of Equation (29) guarantees that the shifting traffic $\triangle M_{h,n}$ is less than NP_n's receiving traffic

threshold, to avoid the NP being heavily loaded (similar to Equation (6)).

$$\Delta M_{h,n} = \left(\tilde{L}_{pars} - L_n\right) \times M \quad n \in \{1 \ldots N\} \tag{28}$$

$$\text{Subject to} \quad \Delta M_{h,n} \leq (L_{HL} - L_n) \times M \tag{29}$$

(4) *Cell-specific handover offset adjustments*

On the basis of the required shifting traffic, BS_h offloads relevant users to *Partner s* (*Partner s* can be *public partner* or *non-public partner* in BS_h's cluster), by adjusting the cell-specific $HO_{off}(h,s)$. Then, U_k *in* BS_h will be offloaded to *Partner s*, if its $RSRP_{k,h}$ from BS_h and $RSRP_{k,s}$ from *Partner s* meet the handover condition (30) [9,29]:

$$RSRP_{k,s} + HO_{off}(h,s) > RSRP_{k,h} + HO_{hys} \tag{30}$$

where HO_{hys} is the handover hysteresis needed to tackle the ping-pong handover (where a user is handed over to a partner and then it is handed back to the cluster head). The simulator in Section 5 sets $HO_{hys} = 2$ db [30]. Because of users random channel condition, BS_h adjusts $HO_{off}(h,s)$ with the step-size θ ($HO_{off}(h,s) = HO_{off}(h,s) + \theta$) to offload users, until the number of their released subcarriers reaches the required shifting traffic or $HO_{off}(h,s)$ reaches the maximum handover offset HO_{off}^{max}.

4.3. Signalling load and complexity

This subsection analyses the signalling load of cooperative traffic shifting from its process shown in Figure 4. In the inter-cluster cooperation, RLRM requires to exchange the actual load or relative load between cells. In the intra-cluster cooperation, each cluster head calculates the shifting traffic on the basis of the actual load/relative load, which was obtained in the inter-cluster cooperation. Meanwhile, a cluster head estimates the PP's LB subcarriers on the basis of the relative load without extra information exchanges.

In the inter-cluster cooperation, the complexity of RLRM is $H \times O(H^2)$ to calculate PP_p's relative load/s towards H different cluster heads. In the intra-cluster cooperation, the complexity of calculating the average load of BS_h's partners is $O(N+P)$, and the complexity of calculating the shifting traffic/s of P PPs and N NPs is $2 \times P \times O(1) + 2 \times N \times O(1)$. Hence, the complexity of the intra-cluster cooperation is $O(N+P) + (2N+2P) \times O(1)$. Because few schemes consider the coordination of multiple hot-spot BSs, RLRM consumes requires extra complexity for PP than MLB schemes without coordination mechanism. The complexity of intra-cluster cooperation is similar with that in [8,9], which is $3P \times O(1) + 3N \times O(1)$ under P PPs and N NPs.

5. SIMULATIONS

To evaluate the proposed scheme, a downlink system-level OFDMA simulator is designed on the basis of [3,9,25], for which the key parameters are shown in Table I. The simulator generates three hot-spot areas, including 13 hot-spot cells, as shown in Figure 7.

The cluster-based cooperative LB scheme includes user-vote-assisted clustering algorithm and cooperative traffic shifting algorithm. They are analysed in Sections 5.1 and 5.2, respectively.

5.1. User-vote-assisted clustering

Section 5.1 simulates the proposed user-vote-assisted clustering algorithm. Meanwhile, in its traffic shifting stage, Section 5.1 refers the principle in [8] to adjust HO_{off} between the cluster head and each *partner* in its cluster, as shown in Equations (31a) and (31b). Then, cluster head's edge users will shift to partners.

$$\text{For } PP\ p\ p \in \{1 \ldots P\}: \quad HO_{off}(h,p) = f(L_h - L_p) \Rightarrow (L_h - L_p) \times HO_{off}^{max} \tag{31a}$$

$$\text{For } NP\ n\ n \in \{1 \ldots N\}: \quad HO_{off}(h,n) = f(L_h - L_n) \Rightarrow (L_h - L_n) \times HO_{off}^{max} \tag{31b}$$

where L_p and L_n are the actual load of PP_p and NP_n, respectively; L_h is the actual load of BS_h; and HO_{off}^{max} is the maximum handover offset.

First, Figure 8 evaluates the performance of the proposed user-vote-assisted clustering algorithm in dealing with the virtual partner problem. The maximum number of partner in each cluster is set to *one*. The *load-based clustering* algorithm, which selects partner on the basis of the neighbouring cell's load, is simulated for comparison. Specifically, in *load-based clustering* algorithm, the cluster head selects *one lowest load* neighbouring cell as partner, and then, the cluster head adjusts its HO_{off} with this partner on the basis of their actual load difference, as shown in Equations (31a) and (31b).

Call blocking probability is a widely used LB performance indicator [6–9] because the more balanced load is reached, the more readily new call users can achieve access to the hot-spot cell. The proposed algorithm has lower call blocking probability than a load-based clustering algorithm. In Figure 8, the proposed algorithm further reduces blocking probability by nearly 1%, compared with the

Table I. Simulator parameters.

Parameter	Value
Subcarrier and total band	Subcarrier: 15 KHz; Total: 5 MHz
Frequency	2 GHz
Inter-site distance	500 m
Log-normal shadow fading	Standard deviation: 8 dB
Downlink path-loss model	37.6 $lg\,(r) + 128.1$, *r-km*
Antenna pattern	$A(\theta) = -\min\{12(\theta/\theta_{3\,\mathrm{dB}})^2, A_m\}$ $\theta_{3\,\mathrm{dB}} = 70°, A_m = 20$ dB
Antenna gain	14 dBi
Total BS transmit power	43 dBm
User mobility	Speed 5 m/s, random direction
Scheduler	Max C/I
Physical Resource Blocks (PRB)	Total 25 PRB (12 subcarriers per PRB)
Q_{step}	0.1 (Equation (2))
Q_s	0.01 (Equation (9b))
T_{crit}	5000 ms
Load measurement period	400 ms (Section 2)
Traffic model	Constant 64 kbps/user; users Poisson arrive
Handover execution time	250 ms
Maximum HO_{off} HO_{off}^{max}	9 dB
Handover time-to-trigger	320 ms
HO_{off} adjustments step-size	1 dB (Section 4.2)

BS, base station.

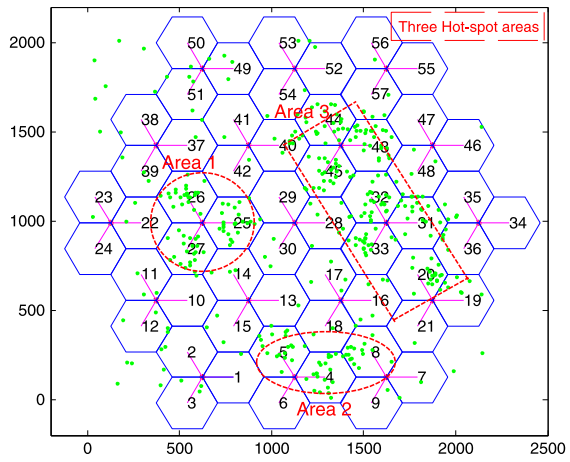


Figure 7. Cellular layout and three hot-spot areas (unit: *meter*).

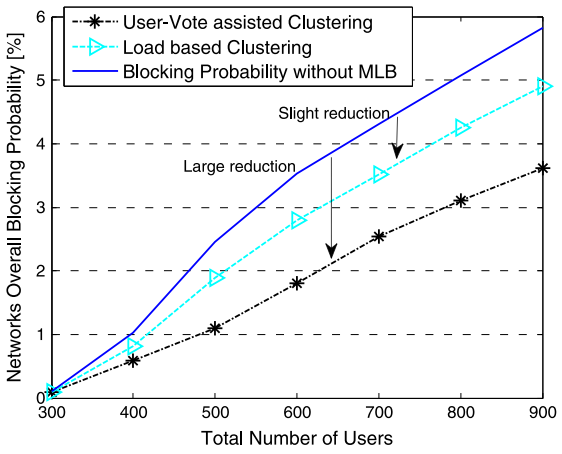


Figure 8. User-vote model effect on overall call blocking probability (one partner).

load-based clustering algorithm. Therefore, the user-vote-assisted clustering algorithm outperforms the conventional load-based clustering algorithm because it can address the virtual partner problem.

Figure 9 further evaluates the user-vote-assisted clustering algorithm by studying the networks overall call blocking probability under different cluster sizes. The proposed algorithm can select the highest priority partner to shift traffic most effectively. The blocking probability can be further reduced if more high priority neighbouring cells are chosen as partners, but the reduction is slight when the number of partners in each cluster goes beyond two.

The traffic shifting stage requires HO_{off} adjustments and frequent LB-related information exchanges [2]. Figure 10 compares the number of HO_{off} adjustments in the user-vote two-partner cluster and that in the typical MLB scheme of [8]. The *ratio of number of* HO_{off} *adjustments* is equal to $\frac{\textit{Number of } HO_{off} \textit{ adjustments in user-vote two-partner cluster}}{\textit{Number of } HO_{off} \textit{ adjustments in typical MLB [8]}}$. The number of HO_{off} adjustments in our proposed user-vote two-partner cluster is much less than the MLB scheme

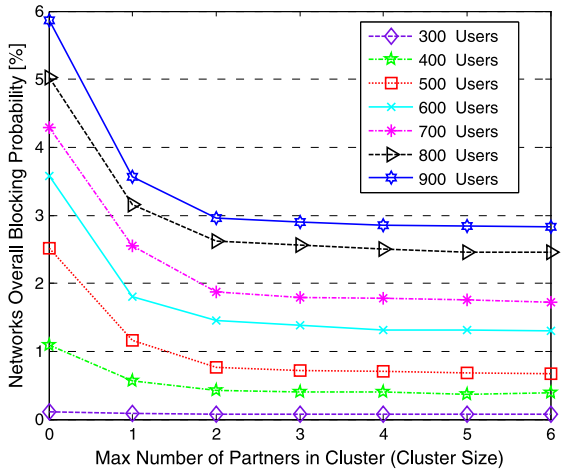


Figure 9. Effect of cluster size on overall blocking probability.

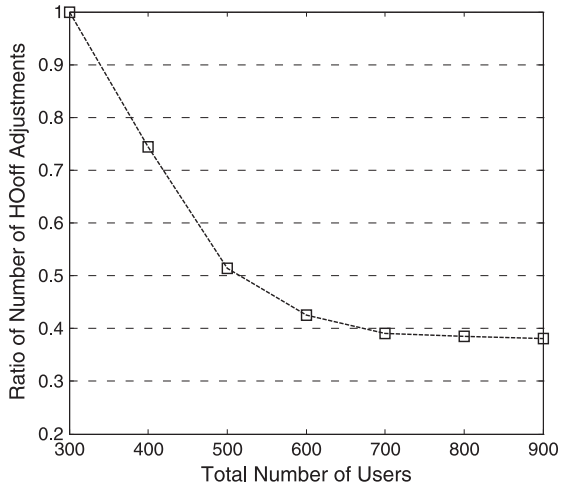


Figure 10. Ratio of number of HO_{off} adjustments.

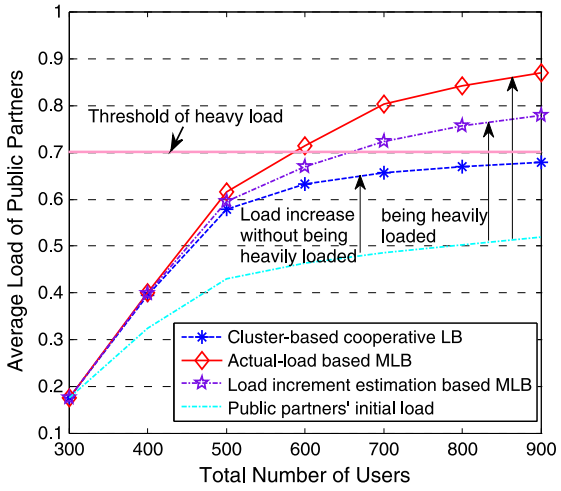


Figure 11. Public partners' average load comparison.

of [8]. For example, the two-partner cluster can reduce up to 60% HO_{off} adjustments. From Figures 9 and 10, the proposed algorithm shows that the two best partners can reach a similar LB performance as choosing three or more partners. In addition, a two-partner cluster can reduce the unnecessary HO_{off} adjustments. On the basis of this, we can conclude that the appropriate cluster size is to have one cluster head with *two partners*.

In summary, Figures 8–10 demonstrate that the proposed clustering algorithm can deal with the virtual partner problem. They also show that selecting a small number of partners (two partners) reaches a good LB performance and improves the clustering efficiency.

5.2. Cooperative traffic shifting

From Figure 7 and 9, Table II shows that 13 cluster heads employ user-vote-assisted clustering algorithm to select their *two* best neighbouring cells as partners. Then, there are eight PPs denoted by *. This subsection evaluates the *cooperative traffic shifting*, including its two key mechanisms: (i) inter-cluster cooperation: RLRM; (ii) intra-cluster cooperation: traffic offloading optimisation.

First, to evaluate the proposed RLRM in addressing the aggravating load problem, the *load increment estimation-based MLB* scheme and the *actual-load-based MLB* scheme are simulated under the same clusters structure of Table II. (MLB schemes, such as [8–12,16], do not analyse the coordination of multiple hot-spot cells' shifting traffic to a PP, and in these schemes, the PP does not report its relative load). Figure 11 shows the average load of PPs after traffic shifting. The actual-load-based MLB scheme results in many heavily loaded PPs. The average load of PPs in the load increment estimation-based MLB scheme is lower than that in the actual-load-based MLB scheme. But the load increment estimation-based MLB may still result in heavily loaded PP because a hot-spot cell cannot control other cells' shifting traffic to the PP. While using the cluster-based cooperative LB scheme, the average load of PPs is always lower than the threshold of heavy load L_{HL}. This is because the relative load coordinates multiple clusters' traffic shifting requests and the partner's idle spectrum available.

Figure 12 depicts the average load of cluster heads after traffic shifting. This evaluates RLRM performance in using the PP's idle spectrum to reduce multiple cluster heads' load. The autonomic MLB scheme [13] is simulated for comparison. As discussed in Section 1, the autonomic MLB module is equipped in each cell to control the traffic shifting, and a lightly loaded cell can share the load of only one cluster head at a time; thus, it can address the appearance of a PP. The proposed scheme has a better capability to reduce cluster heads' load than the autonomic MLB scheme. For example, our scheme can further reduce nearly 10% load, under scenarios with 500 to 900 users.

Table II. Partner selection in each cluster.

Cluster head	Partner	Cluster head	Partner	Cluster head	Partner
Cell4	Cell9 *Cell18	Cell25	*Cell14 *Cell42	Cell32	*Cell28 *Cell48
Cell5	Cell1 *Cell18	Cell26	*Cell22 *Cell42	Cell33	*Cell28 Cell17
Cell8	Cell21 *Cell18	Cell27	*Cell14 *Cell22	Cell43	*Cell48 *Cell57
Cell20	Cell19 Cell36	Cell31	Cell35 *Cell48	Cell44	*Cell40 *Cell57
				Cell45	*Cell40 *Cell28

*Public partner.

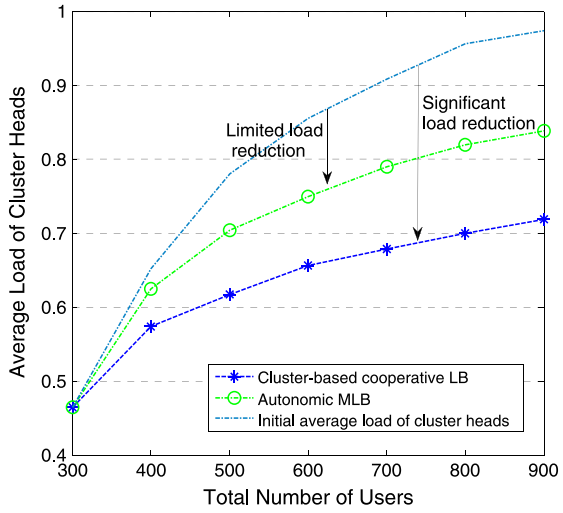


Figure 12. Average load of cluster heads comparison.

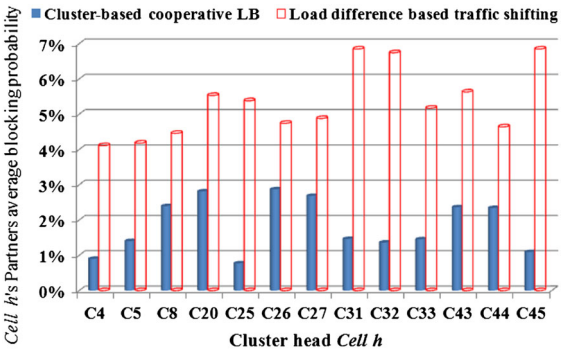


Figure 13. Average call blocking probability of each cluster head's partners.

This is because RLRM allows the appearance of PP and RLRM efficiently pre-allocates the PP's idle spectrum to each cluster head, thus balancing the load without creating a heavily loaded PP.

Finally, this paper evaluates the proposed *traffic offloading optimisation algorithm* performance as compared with the load difference-based traffic shifting scheme. Figure 13 shows the average call blocking probability of each cluster head's partners after receiving traffic, with 900 users in the networks. The proposed cluster-based cooperative LB scheme has much lower call blocking probability than that in the load difference-based traffic shifting scheme.

Load difference-based traffic shifting scheme introduction

As introduced in Section 4.2, the cluster head BS_h tries to release $\triangle M_h$ subcarriers, which is flexible according to different LB objectives. This simulator assumes that BS_h's LB objective L_h^* equals the average load of its cluster, namely $L_h^* = \left(L_h + \sum_{n=1}^{N} L_n + \sum_{p=1}^{P} R_{p,h}\right) / (1 + N + P)$. Therefore, BS_h's load reduction ΔL_h equals $L_h - L_h^*$. BS_h's releasing subcarriers ΔM_h can be expressed as

$$\begin{aligned} \Delta M_h &= M \times \Delta L_h \\ &= M \times \left(L_h - \frac{L_h + \sum_{n=1}^{N} L_n + \sum_{p=1}^{P} R_{p,h}}{1 + N + P} \right) \end{aligned} \tag{32}$$

Under the condition of releasing $\triangle M_h$ subcarriers, the proposed relative load-based traffic shifting calculation mechanism refers to Section 4.2.

Because the traffic shifting direction is from a hot-spot cell to each partner, MLB schemes in [8–10,12] calculate the shifting traffic and adjust HO_{off} between the hot-spot cell and each partner on the basis of their load difference. However, the simulation comparison cannot directly use the equations in these load difference schemes [8–10,12]. It is because their equations cannot ensure a pre-defined overall load reduction $\triangle L_h$ of the cluster head, under two or more partners (Equation (31) shows that the shifting traffic and HO_{off} function have no constraint of cluster head's overall load reduction).

The simulation tries to avoid the cluster head having different load reduction objectives, in the conventional 'load difference' scheme and our 'cluster-based cooperative LB' scheme. Hence, this paper follows the load difference principle and designs the 'load difference-based traffic shifting' scheme, in which the cluster head always has a certain overall load reduction $\triangle L_h$ under different numbers of partners. In this scheme, for a particular partner, the shifting traffic $\triangle M_{h,p}$ or $\triangle M_{h,n}$ is based on the actual load between BS_h and this partner as follows:

- Shifting traffic from BS_h to PP_p $p \in \{1 \ldots P\}$: $\Delta M_{h,p} = \frac{(L_h - L_p)}{\sum_{\breve{p}=1}^{P}(L_h - L_{\breve{p}}) + \sum_{\breve{n}=1}^{N}(L_h - L_{\breve{n}})} \times \Delta M_h$
- Shifting traffic from BS_h to NP_n $n \in \{1 \ldots N\}$: $\Delta M_{h,n} = \frac{(L_h - L_n)}{\sum_{\breve{p}=1}^{P}(L_h - L_{\breve{p}}) + \sum_{\breve{n}=1}^{N}(L_h - L_{\breve{n}})} \times \Delta M_h$

where L_h is the actual load of the cluster head and L_p and L_n are the actual load of PP_p and NP_n, respectively. $\triangle M_h$ is BS_h's total releasing subcarriers calculated in Equation (32). In addition, the two schemes in Figure 13 have the same cluster structure as shown in Table II.

6. CONCLUSIONS

This paper has proposed a self-organising cluster-based cooperative LB scheme for OFDMA cellular networks. In the clustering stage, the hot-spot cell employs a user-vote model, which considers the user's channel condition, for selecting suitable partners to provide good service for shifting users. Simulation results show that the user-vote-assisted clustering can address the virtual partner problem. It also can select two best partner cells to efficiently balance the load. After cell clustering, this paper develops a new cooperative traffic shifting algorithm that involves inter-cluster cooperation and intra-cluster cooperation. In the inter-cluster cooperation, the PP employs the RLRM to coordinate the traffic shifting requests from multiple clusters to address the aggravating load problem. In the intra-cluster cooperation, the traffic offloading optimisation algorithm minimises the partners' average call blocking probability in each cluster. Simulation results show that the proposed scheme can keep the PP's load lower than the heavily loaded threshold. The scheme also achieves partners' average call blocking probability reduction than the load difference-based traffic shifting scheme.

APPENDIX A

A1. The analysis of η=4 (in user-vote model of Equation (2))

In Equation (2), U_k (*User k*) tries to set an appropriate $SINR_{k,h}/\eta$ to identify cell edge user and to calculate its *vote* towards neighbouring BS_i. Hence, the 3 dB cell edge user identification criterion of [31] is used, as

$$(RSRP_{k,h})_{\mathrm{dB}} - (RSRP_{k,i})_{\mathrm{dB}} \leq 3\,\mathrm{dB} \Rightarrow \frac{(RSRP_{k,h})_{linear}}{(RSRP_{k,i})_{linear}} \leq 2 \tag{A1}$$

where $RSRP_{k,h}$ is from its serving BS_h, and $RSRP_{k,i}$ is from neighbouring BS_i. The 3 dB denotes that their RSRP ratio is two times in linear format. Then, we analyse its SINR relationship. The RSRP and SINR in Equations (A2) and (A3) are in the linear format.

$$SINR_{k,h} \geq \frac{RSRP_{k,h}}{RSRP_{k,i} + \sum_{\bar{i}=1,\bar{i}\neq i}^{I} RSRP_{k,\bar{i}}} = \frac{2 \times RSRP_{k,i}}{0.5 \times RSRP_{k,h} + \sum_{\bar{i}=1,\bar{i}\neq i}^{I} RSRP_{k,\bar{i}}} \tag{A2}$$

$$\approx 4 \times \frac{RSRP_{k,i}}{RSRP_{k,h} + \sum_{\bar{i}=1,\bar{i}\neq i}^{I} RSRP_{k,\bar{i}}} = 4 \times SINR_{k,i}^{est} \tag{A3}$$

where $SINR_{k,h}$ is U_k's serving SINR from BS_h and $SINR_{k,i}^{est}$ is U_k's SINR estimation towards BS_i.

Equation (A2) sets '$\geq$' because $RSRP_{k,i} + \sum_{\bar{i}=1,\bar{i}\neq i}^{I} RSRP_{k,\bar{i}}$ is the theoretical heaviest overall interference of $SINR_{k,h}$. In Equation (A3), '$\approx$' denotes approximately because if U_k is shifted, $RSRP_{k,h}$ from the cluster head becomes the heaviest interference, compared with $RSRP_{k,\bar{i}}$ from other neighbouring BSs. Therefore, $\eta = 4$ is a suitable value in the user-vote model to calculate vote.

REFERENCES

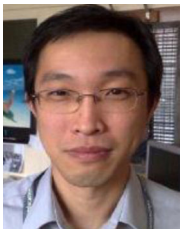
1. Lee BG, Park D, Seo H. *Wireless Communications Resource Management*. John Wiley & Sons: Singapore, 2009.
2. 3GPP TS 36.300. E-UTRA and E-UTRAN overall description V9.5.0, September 2010.
3. 3GPP TR 25.814. Physical layer aspects for evolved universal terrestrial radio access V7.1.0, September 2006.
4. Hernández A, Guío I, Valdovinos A. Radio resource allocation for interference management in mobile broadband OFDMA based networks. *Wiley Wireless Communications and Mobile Computing* 2010; **10**(11): 1409–1430.
5. Ma D, Ma M. Proactive load balancing with admission control for heterogeneous overlay networks. *Wiley Wireless Communications and Mobile Computing* 2011; DOI: 10.1002/wcm.1224.
6. Tonguz OK, Yanmaz E. The mathematical theory of dynamic load balancing in cellular networks. *IEEE Transactions on Mobile Computing* 2008; **7**(12): 1504–1518.
7. Jiang H, Rappaport SS. Channel borrowing without locking for sectorized cellular communications. *IEEE Transactions on Vehicular Technology* 1994; **43**(4): 1067–1077.
8. Nasri R, Altman Z. Handover adaptation for dynamic load balancing in 3GPP long term evolution systems, In *Proceedings of International Conference on Advances in Mobile Computing and Miltimedia (MoMM)*, Jakarta, December 2007; 145–153.

9. Kwan R, Arnott R, Paterson R, Trivisonno R, Kubota M. On mobility load balancing for LTE systems, In *Proceedings of IEEE VTC-Fall*, Ottawa, Canada, September 2010; 1–5.
10. Yang Y, Li P, Chen X, Wang W. A high-efficient algorithm of mobile load balancing in LTE system, In *Proceedings of IEEE VTC-Fall*, Quebec, Canada, September 2012; 1–5.
11. Wei Y, Peng M. A mobility load balancing optimization method for hybrid architecture in self-organizing network, In *Proceedings of IET ICCTA*, Beijing, China, October 2011; 828–832.
12. Lv W, Li W, Zhang H, Liu Y. Distributed mobility load balancing with RRM in LTE, In *Proceedings of IEEE IC-BNMT*, Beijing, China, October 2010; 457–461.
13. Zhang H, Qiu X, Meng L, Zhang X. Design of distributed and autonomic load balancing for self-organization LTE, In *Proceedings of IEEE VTC-Fall*, Ottawa, Canada, September 2010; 1–5.
14. Zhang H, Qiu X, Meng L, Zhang X. Achieving distributed load balancing in self-organizing LTE radio access network with autonomic network management, In *Proceedings of IEEE GLOBECOM*, Miami, USA, December 2010; 454–459.
15. Hu H, Zhang J, Zheng X, Yang Y, Wu P. Self-configuration and self-optimization for LTE networks. *IEEE Communications Magazine* 2010; **48**(2): 94–100.
16. Rodríguez J, Bandera IDL, Munoz P, Barco R. Load balancing in a realistic urban scenario for LTE networks, In *Proceedings of IEEE VTC-Spring*, Budapest, Hungary, May 2011; 1–5.
17. Lobinger A, Stefanski S, Jansen T, Balan I. Load balancing in downlink LTE self-optimizing networks, In *Proceedings of IEEE VTC-Spring*, Taipei, May 2010; 1–5.
18. Suga J, Kojima Y, Okuda M. Centralized mobility load balancing scheme in LTE systems, In *Proceedings of ISWCS*, Aachen, Germany, November 2011; 306–310.
19. Jansen T, Balan I, Turk J, Moerman I, Kurner T. Handover parameter optimization in LTE self-organizing networks, In *Proceedings of IEEE VTC-Fall*, Ottawa, Canada, September 2010; 1–5.
20. Xu L, Chen Y, Schormans J, Cuthbert L, Zhang T. User-vote assisted self-organizing load balancing for OFDMA cellular systems, In *Proceedings of IEEE PIMRC*, Toronto, Canada, September 2011; 217–221.
21. Xu L, Chen Y, Chai KK, Zhang T, Schormans J, Cuthbert L. Cooperative load balancing for OFDMA cellular networks, In *Proceedings of European Wireless*, Poznan, Poland, April 2012; 1–7.
22. 3GPP TS 36.314. Layer 2 – Measurements V10.0.0, December 2010.
23. Wang Y, Zhang P. *Radio Resource Management*. Beijing University of Posts and Telecomm Press: Beijing, 2005.
24. Goldsmith A. *Wireless Communications*. Cambridge University Press: New York, 2005.
25. Ramiro J, Hamied K. *Self-Organizing Networks (SON): Self-Planning, Self-Optimization and Self-Healing for GSM, UMTS and LTE*. John Wiley & Sons: Chichester, 2011.
26. 3GPP TS 36.420. X2 general aspects and principles V10.0.1, March 2011.
27. Tsai YR, Chang CJ. Cooperative information aggregation for distributed estimation in wireless sensor networks. *IEEE Transactions on Signal Processing* 2011; **59**(10): 3876–3888.
28. Hanson MA. Invexity and the Kuhn-Tucker theorem. *Journal of Mathematical Analysis and Applications* 1999; **236**: 594–604.
29. Hwang RH, Chang BJ, Lin YM, Liang YH. Adaptive load-balancing association handoff approach for increasing utilization and improving GoS in mobile WiMAX networks. *Wiley Wireless Communications and Mobile Computing* 2012; **12**(14): 1251–1265.
30. Lee DW, Gil GT, Kim DH. A cost-based adaptive handover hysteresis scheme to minimize the handover failure rate in 3GPP LTE system. *Eurasip Journal on Wireless Communications and Networking* 2010; 2010: Article ID: 750173. DOI: 10.1155/2010/750173.
31. Sawahashi M, Kishiyama Y, Morimoto A, Nishikawa D, Tanno M. Coordinated multipoint transmission/reception techniques for LTE-Advanced. *IEEE Wireless Communications* 2010; **17**(3): 26–34.

AUTHORS' BIOGRAPHIES

Lexi Xu received BSc degree from Southwest Jiaotong University, Chengdu, China, in 2006. He receives MSc degree from Beijing University of Posts and Telecommunications, Beijing, China, in 2009. Since September 2009, he joined School of Electronic Engineering and Computer Science, Queen Mary University of London, where he worked towards the PhD degree and he receives his PhD degree in 2013. From late 2013, he will also join China Unicom to carry on research work. His research interests include cooperative communications, radio resource management and LTE-A

Yue Chen received her BSc in Telecommunications and MSc in Optical Communications from Beijing University of Posts and Telecommunications (BUPT), China, in 1997 and 2000, respectively. She received her PhD in Wireless Communications from Queen Mary University of London (QMUL) in 2003 and became an academic member of staff in School of Electronic Engineering and Computer Science. From 2005 till now, she holds the deputy director role of the Joint Programme between QMUL and BUPT. Her research interests include intelligent radio resource management for wireless networks, scheduling and load balancing optimization, CoMP, cognitive radio, LTE-A and Internet of Things.



Kok Keong Chai received his BEng (Hons), MSc and PhD at University of Hertfordshire, UK, in 1998, 1999 and 2007, respectively. From 2002 to 2008, he was a senior lecturer in Applied Computing at Staffordshire University. He joint School of Electronic and Computer Science, Queen Mary University of London, UK, in 2008 as a joint programme tutor. His current research interests include intelligent radio resource management, load balancing optimisation, LTE-A networks, Internet of Things and intelligent transport systems.

John Schormans joined the Telecommunications Research Group at Queen Mary University of London, where he gained a PhD in 1990. His research interests include the application of probabilistic methods to the analysis, simulation and performance measurement of communications systems and networks, focussing on packet-based technologies. He has published over 120 research papers, supervised 11 PhD students and twice acted as the Editor for Special Editions of IET Communications. He has been a member of the Editorial Board for the journal IET Communications, a BT Short Term Research Fellow and Principal Investigator on two EPSRC projects. He is a co-author of Introduction to ATM Design and Performance, published by Wiley in 1996.

Laurie Cuthbert is a Professor at the School of Electronic Engineering and Computer Sciences at Queen Mary University of London, Dean for China Operations and director of the Intelligent Systems Research Centre in Macao. He has been active in a number of European research activities, leading a task group in RACE R1022 and then work packages in later EU projects. He provided the technical management for the ACTS project IMPACT and in the IST projects SHUFFLE and ADAMANT.