Salhani, Mohamad

Offloading the Small Cells for Load Balancing in UDNs Using the Proactive and the User Transfer Algorithms with Reducing the APs Inter-communications

Published in:
Advanced Information Networking and Applications - Proceedings of the 34th International Conference on Advanced Information Networking and Applications, AINA 2020

DOI:
10.1007/978-3-030-44041-1_81

Published: 01/01/2020

Document Version
Peer-reviewed accepted author manuscript, also known as Final accepted manuscript or Post-print

Please cite the original version:
Salhani, M. (2020). Offloading the Small Cells for Load Balancing in UDNs Using the Proactive and the User Transfer Algorithms with Reducing the APs Inter-communications. In L. Barolli, F. Amato, F. Moscato, T. Enokido, & M. Takizawa (Eds.), *Advanced Information Networking and Applications - Proceedings of the 34th International Conference on Advanced Information Networking and Applications, AINA 2020* (pp. 934-946). (Advances in Intelligent Systems and Computing; Vol. 1151 AISC). Springer. https://doi.org/10.1007/978-3-030-44041-1_81

Offloading the Small Cells for Load Balancing in UDNs Using the Proactive and the User Transfer Algorithms with Reducing the APs Inter-Communications

Mohamad Salhani
Aalto University, Espoo, Finland
mohamad.salhani@aalto.fi

Abstract. In Ultra-Dense Networks, the dense deployment of small cells generates an uneven traffic distribution. This causes network performance degradation. To address this problem, this paper proposes a proactive algorithm along known user transfer and reactive algorithms to construct small-cell clusters and perform load balance across the Access Points (APs). The proactive algorithm distributes the new users, user by user, to the small cells, while the reactive algorithms are triggered when the load of the chosen cluster reaches a predefined threshold. The transfer algorithms offload the small cells by transferring the extra users to the macrocells. The user transfer can occur before or after balancing the load by the reactive algorithm. Moreover, this paper adapts the Design Structure Matrix (DSM) method to balance the load and reduce the APs inter-communications. The results indicate that the proactive algorithm with the transfer algorithms and the DSM method improve the load distribution and significantly reduce the APs inter-communications.

1 Introduction

The wireless data demand has increased explosively in the past decade, as the use of smart devices and applications has significantly increased. To fulfill the heavily growing data demands, the Ultra-Dense Network (UDN) is considered as a promoting solution for the 5G cellular networks [1]. However, due to the users' mobility, the random deployment of small cells and the preference of small cells during the selection and reselection, the load across the APs becomes unbalanced. This causes network performance degradation. When the users move onto overloaded small cells, even if neighboring cells remain underloaded, the deficit of resources in overloaded cell results in handover failures or poor Quality of Service (QoS) requirements, while other neighboring cell resources remain unused. On the other hand, the Design Structure Matrix (DSM) method provides a representation of a complex system that supports innovative solutions to decomposition and integration problems used in the system engineering of products [2]. Actually, the DSM method has not been exploited yet in the previous studies that deal with the Load balancing (LB). In this paper, the DSM method with the proactive and transfer algorithms are adapted to reduce the APs inter-communications and balance the load as well.

To balance the load and improve the performance of cellular networks, the centralized Self-Organized Network (cSON) is a promoting solution to configure and optimize the network [3]. The cSON has many features, like mobility robustness, optimization, Mobility Load Balancing (MLB), interference management, and so on [4]. The MLB algorithm optimizes the handover parameters and achieves LB without affecting the user experience. Thus, it is necessary to study a LBA adapted to various network environments and avoid the load ping-pongs. Moreover, researchers have proposed several solutions to address the LB problem and enhance cellular network performance. The authors in [5] have proposed an MLB algorithm considering constant-traffic users with a fixed threshold to determine overloaded cells in Long Term Evolution (LTE) networks. Nevertheless, owing to the fixed threshold, the algorithm is not able to perform LB adaptive to varying network environments. In [6], the authors have proposed an MLB algorithm considering an adaptive threshold to decide overloaded cells in a small cell network.

The authors in [7] have mathematically proved the balance efficiency of the proposed LBAs based on the overlapping zones between the intersecting small cells. The proposed LBA was small cell cluster-based and intended first to determine the best overlapping zone and then, to select the best user for handover in order to reduce the number of the handovers and improve the UDN performance. Nonetheless, the proposed algorithm was reactive; it is only achieved when the user density of the chosen small-cell cluster reaches a predefined threshold.

Alternatively, the LB, by transferring users to other networks, has not been highlighted enough in the recent studies. *Elgendi et al* [8] have proposed schemes to determine the optimal number of sessions to be transferred from Unlicensed Long Term Evolution (U-LTE) networks to Licensed one (L-LTE) or Wi-Fi networks. The users transfer from programmable Base Stations (BSs) to APs to achieve a win-win outcome for both networks. However, they have focused on the users' velocity and the distance between the user and the BS more than the data offloading. Besides, the proposed schemes transfer a higher number of users. In contrast, the authors in [9] have proposed user transfer algorithms to offload the small cells in UDNs by transferring the extra users to the BSs. The best overlapping zone is first identified and then, the Best Candidate (BC) user is handed over to another AP or BS.

In this paper, we propose a proactive algorithm with the transfer algorithms proposed in [9] to construct small-cell clusters and perform the LB across the APs. For cluster formation, the algorithm considers an overloaded small cell and two neighboring small cells. Thus, the algorithm performs the LB locally and updates Cell Individual Offset (CIO) parameters of the cells. In addition, we employ the DSM method in reducing the APs inter-communications and balancing the load as well.

The rest of this paper is organized as follows: Section II describes the system model and assumptions we made. The LBAs are proposed in Section III followed by the DSM method in Section IV. Section V presents the DSM method with other algorithms. The results are evaluated in Section VI. Section VII concludes the paper.

2 System Model

2.1 System Description

We consider a heterogeneous LTE network composed of a set of macrocells; evolved Nodes B (eNB) and small cells, *N*, and a set of users, *U*, [6], [7]. We consider the UDN small cells with overlapping zones (Z_1, Z_2, Z_3 and Z_4) and each set of small cells constitutes a so-called cluster. In the simulation model, we consider a cluster consists of three intersecting small-cells, as done in [7], as depicted in Fig. 1. The cells interconnect with each other via X2 interface. This allows them to perform the needed functionalities such as handovers, load management, and so on. Therefore, the users can move seamlessly among the cells. To optimize the parameters in the network, a cSON subsystem is considered [4]. The cells are connected to the cSON subsystem via S1 interface. The cSON subsystem collects the required load-related information from the network and optimizes the parameters of the cells to perform the LB process.

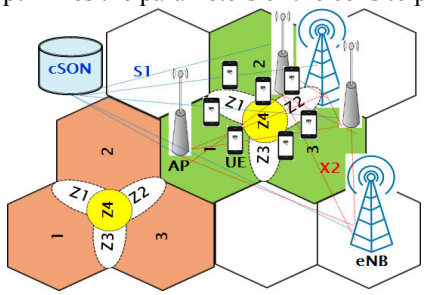

Fig. 1. System model with a cSON.

2.2 Small Cells Load

To measure the load in each cluster, the average Resource Block Utilization Ratio (*RBUR*) is calculated from the Physical Resource Blocks (*PRBs*) allocation information, [6]. For a given time duration, *T,* the small cell load, ρ_i, of cell *i* at time *t*, is given as

$$\rho_i^t = \frac{1}{T.N_{PRB}} \sum_{\tau \in (t-T,\ t)} RB_i^{\tau} \quad \textbf{(1)}$$

where N_{PRB} and *RB(i, j)* denote the total *PRBs* and the total allocated *PRBs* for all the users, *U*, in cell *i*, respectively. Hence, the average cluster load, *ACL*, is expressed as

$$ACL = (\sum_{i=1}^{m} \rho_i) / m \quad \textbf{(2)}$$

where *m* is the number of the small cells constituting the cluster. To identify overloaded, balanced and underloaded small cells in each cluster, two adaptive thresholds are introduced; upper and lower thresholds, δ_1, δ_2, [7] as follows

$$\delta_1 = ACL + \alpha \times ACL\,;\ \ \delta_2 = ACL - \alpha \times ACL \quad \textbf{(3)}$$

where α is the tolerance parameter, which controls the balance zone's width. A small

α requires many handovers to reach the LB, and vice-versa. α is set to 0.05, [7].

2.3 Handover Procedure

In this paper, A3 and A4 event measurements are used to trigger a handover and select the BCs for handovers, and the Reference Signal Received Power (*RSRP*) is assumed reporting signal quality for measurements [6]. Actually, event A3 is widely used for triggering handovers in wireless networks [10]. In that way, event A3 is triggered and the users report the measurement results to the serving cell when the signal of a neighboring cell in a cluster is offset better than that of the serving cell. If the event A3 triggering criteria remains satisfied for longer than the Time to Trigger (TTT), the cell decides to trigger a handover. The event A3 measurement is reported if the following condition is satisfied [6]:

$$Mn + Ofn + Ocn - Hyst > Mp + Ofp + Ocp + off \quad \textbf{(4)}$$

where *Mn* and *Mp* denote the average *RSRP* values. *Ofn* and *Ofp* are the frequency-specific offsets. *Ocn* and *Ocp* are the cell individual offsets for the target and the serving cells, respectively. *Hyst* is the hysteresis parameter. *Off* is the A3 event offset between the serving and the target cells. The cSON performs the LB by shifting the users in the overloaded cells to the underloaded cells. However, to balance the load, the system needs information about the edge users' distribution. For that, the event A4 is used. All the cells share the users' information with the cSON. The condition for triggering event A4 is expressed as [6],

$$Mn + Ofn + Ocn - Hyst > Thresh \quad \textbf{(5)}$$

where *Thresh* is event A4's threshold. The users that satisfy this condition report measurements for the serving and neighboring cell within the cluster. In this regard, each cell makes a set of edge users based on A4 event reports. Then the cSON collects all the edge-users' information from all cells. The LBA selects the BC edge-user and hands over it to the best target cell according to the chosen LB scheme.

3 Load Balancing Algorithms (LBAs)

In this section, we present the different LBAs that can be used to balance the load.

3.1 Proactive Algorithm (Pro)

The Pro distributes the new users to the covering APs. This algorithm is always on standby and ready to be triggered each time a new user enters the network. For each new user, the algorithm selects the least loaded AP. If the new user is a constrained user, this user will be accepted by the chosen AP even if this results in exceeding the ρ_{th} limit. A constrained user is a DSM user that is communicating with another user located in the same cluster. The new users' distribution lasts until the user density; D of the chosen cluster reaches the user density threshold, D_{th}.

3.2 Reactive Algorithm (Rea)

The reactive algorithm (Rea), which proposed in [7], is adapted again in this paper to balance the load. This algorithm is only triggered when the user density reaches D_{th}. To achieve the Rea, three approaches are suggested based on the chosen overlapping zones. In the Common Zone (CZ) approach, the load is only balanced via the users that are located in the CZ between the three overlapping small cells; zone four (Z_4), as shown in Fig. 1. The second approach is the so-called Worst Zone (WZ) approach. The LB is performed in the WZ, which has the smallest value of the Jain's fairness index, β (explained later). The balance efficiency of this approach has been mathematically proven in [7]. The third approach is the Mixed Approach (MA), which starts balancing the load in the CZ and then, it transits into the WZ with or without returning to the CZ.

To balance the load using the Rea, the cluster with the highest density is identified and then the overlapping zone and the BC for handover. It **first** compares the density of the cluster, D, with the highest density to the density threshold, D_{th}. If the user density does not exceed the D_{th}, the algorithm is stopped. Otherwise, it sets the user's load, $RBUR_j$ of each $user_j$, its zone and α. Next, the algorithm calculates the load of each AP, ρ_i, and the ACL. Meanwhile, the algorithm determines the AP state by the transfer policy. This policy verifies which AP must exclude a user (overloaded AP) and which one must include this user (underloaded AP). For that, two thresholds, δ_1 and δ_2 are needed. In the **second** step, the algorithm checks if there is at least one overloaded AP within the cluster with the highest user density (cluster of first order). If not, the algorithm transits into the cluster of second or third order successively and rechecks the user density condition. If this condition is not satisfied in these three clusters, the algorithm is stopped. Otherwise, the algorithm determines the Jain's fairness index *(β)* [11] as follows

$$\beta = \left(\sum_{i=1}^{n} \rho_i\right)^2 \Big/ \left(n \times \sum_{i=1}^{n} \rho_i^2\right) \quad \textbf{(6)}$$

where n is the number of the small cells that overlap on the zone in question, i.e., each overlapping zone has its own β. The **third** step is to apply the selection policy to identify the BC for handover. For that, the difference (Δ) between the load of the chosen overloaded AP and the ACL is calculated by

$$\Delta = \rho_{overloaded\_AP} - ACL \quad \textbf{(7)}$$

Of all the users located in the overlapping zone in question and connected to the chosen overloaded AP, the BC is the one for which the difference of the user's load and Δ has the smallest absolute value as follows

$$BC_j = \left| RBUR_j - \Delta \right| \quad \textbf{(8)}$$

Note that some constrained users may be excluded from any handovers, as explained later. The **fourth** step is to calculate the new β if the BC is handed-over. This is done by the distribution policy to ensure that the expected handover will definitely enhance the balance before achieving the handover. Thus, the handover occurs if and only if $\beta_{new} > \beta_{old}$. Otherwise, the algorithm transits into the next target zone. The target zone is one of the overlapping zones, which changes or not according to the chosen LB scheme. For instance, the target zone in the WZ approach is the one that has the

smallest value of β. The **fifth** step is to check if there is still an overloaded AP and if the balance improvement is still valid. If so, the LB enhancement is evaluated again in the new target zone and so on. Otherwise, the algorithm is stopped.

3.3 User Transfer Algorithms

Once the users are distributed to the APs by the Pro, the transfer algorithms start offloading the small cells. In this regard, two transfer algorithms are proposed [9].

3.3.1 Transfer_After Algorithm (TAA)

The TAA is composed of two stages. The first one is the balance stage achieved by the Rea. The second one is the transfer stage. Therefore, the TAA has the same first steps of the Rea; however, when there are no more balance improvements, the transfer stage is initialized. The algorithm checks if at least one of the APs exceeds the ρ_{th}. If not, the algorithm is stopped. Otherwise, the second step is to achieve the new selection policy to determine the BC for transfer. Hence, the algorithm first calculates the new delta as a difference between the most overloaded AP and the ρ_{th} as follows,

$$\Delta_{new} = \rho_{most\_overloaded\_AP} - \rho_{th} \quad \textbf{(9)}$$

Second, the BC value $BC_{(j)}$ is calculated for each user connected to the chosen AP as a difference between the user's load and the new delta as follows,

$$BC_{(j)} = \mathrm{RBUR}_{(j)} - \Delta_{new} \quad \textbf{(10)}$$

Of all the users connected to the AP in question, the BC is the one for which the $BC_{(j)}$ has the smallest positive value. Otherwise, the BC is the one that has the smallest negative value in case all the values of $BC_{(j)}$ are negative. The transfer from the chosen AP is repeated until the ρ_i becomes less than or equal to ρ_{th}. In the third step, the algorithm identifies the next most overloaded AP and repeats the second step. When all the APs have checked and there is no transfer anymore, the TAA is stopped.

3.3.2 Transfer_Before Algorithm (TBA)

The TBA is similar to the TAA; however, the transfer stage is initialized as a first step for each AP that has a load exceeding the ρ_{th}. When the load of each AP does not exceed the ρ_{th} or if there are no more users available for transfer, the balance stage starts by calling the Rea to continue the LB task as usual.

4 DSM Method

The DSM method has been proposed in [12] and is employed again in this paper to reduce the APs inter-communications and balance the load. This method deals with partitioning of graphs to realize a cooperation between the nodes with respect to parallel, consecutive and coupled tasks. To explain this method, Fig. 2 shows a graph

composed of six nodes. The details of the DSM method are mentioned in [12].

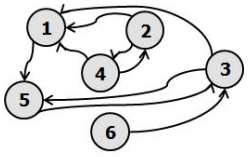

Fig. 2. The graph of the nodes

To redistribute these nodes on two switches/APs, with respect to the type of tasks (serial, parallel), four matrices are needed. These matrices are concerned with the direct or indirect connections between the nodes. Then, the parallel and in series tasks are deduced. Next, the new groups are determined after reordering them. The final matrix identifies the relationships between the new groups, i.e., the parallel, serial tasks and the inter-group tasks. For instance, assuming the new groups, after applying the DSM method, become as follows: G_1= (1, 3, 5), G_2=(2, 4) and G_3=(6). Assuming we have two switches, the groups can be partitioned as follows: G_1=(1, 3, 5) and G_2=(2, 4, 6). This distribution can be refined to reduce the inter-communications between the switches. For that, the replacement gain (G) for each node is introduced. It is the difference between the number of the connections of a node with the other groups and the number of the connections of this node with the nodes existing in its group. Fig. 3 shows the gain matrix before and after the refinement. The refinement process is concerned with the nodes with positive gains (node 6 in G_2 with G=1). Accordingly, node 6 must be replaced by node 1, which has the biggest gain within G_1 with G=0. To evaluate the replacement performance, the load index, τ is introduced. The τ is defined as the ratio of the number of inter-group connections, N_i, to the total number of interconnections of all the nodes, N_t: $\tau = N_i / N_t$ **(11)**. After the replacement, the new nodes distribution becomes G1=(3, 5, 6) and G2=(1, 2, 4). The initial value, τ_{inital} was 3/9 and the final value, τ_{final} after the refinement becomes 2/9. As a result, the switches inter-communications are reduced.

	G1			G2		
	1	3	5	2	4	6
G1	2	3	3	1	1	1
G2	2	1	0	2	2	0
Gains	0	-2	-3	-1	-1	1

	G1			G2		
	3	5	6	1	2	4
G1	3	2	1	2	0	0
G2	1	1	0	2	3	3
Gains	-2	-1	-1	0	-3	-3

Fig. 3. Users' distribution after applying DSM (a) and users' distribution after refinement (b).

The question is how the DSM method and the refinement can be exploited with the proactive and the transfer algorithms. In fact, the Rea is triggered when the user density reaches D_{th} and the DSM method is applied after distributing the users by the Pro. The transfer algorithms are also triggered after distributing the users by the Pro. At that time, the users have already been connected to the APs and each AP has been constituted a group of some connected users. Hence, the required aim is to apply the refinement. Actually, to use the DSM method, either, the replacement stage is first applied and then, the balancing stage is achieved by the Rea. This policy is performed by the *DSM_first algorithm (DSMf)*. Or, the DSM constraints are respected by the LBA during the selection policy. This latter is achieved by the *DSM_included algorithm (DSMi)*. In both policies, the DSM constraints impose that the selection of a

user to be replaced is only allowed if the number of hops of the user's connection is kept constant or reduced from three to two hops.

5 DSM Algorithms (DSMAs) with Other Algorithms

To apply the DSM method, two DSM algorithms (DSMAs) are used [12].

5.1 DSM Algorithms

The DSMi is a Rea; however, the replacement gain, *G* is respected, i.e., the chosen user in not selected to be the BC and thereby handed-over, if this handover will increase *G*. In contrast, the DSMf first reduces the APs inter-communications and then, it starts the LB using the Rea as follows. The **first and second** steps of the DSMf are the same as the Rea. Next, the gain matrix for each AP is computed. In the **third** step, the algorithm searches, in the gain matrix of the most loaded AP, for a user that has the highest positive gain. This means this user is communicating with another user (its partner), which is connected to another AP in the same cluster. In the **fourth** step, the algorithm checks the coverage condition: the AP of the candidate user and the AP of the partner should cover the two users. Thus, the algorithm replaces the selected user by the BC. The selected user is a DSM user, has the highest load, is connected to the most loaded AP and has the highest positive gain. Instead, the BC is the one that is connected to the partner's AP, is located in the same zone of the selected user and has the lowest load. If the coverage condition is not met, the **fifth** step is to check if there are other DSM users with a positive gain and connected to the most loaded AP. If so, a new DSM user is selected and so on. Otherwise, the algorithm transits into the next most loaded AP and repeats the third step. Once all the APs are checked and the replacements are over, the DSMf calls the Rea to continue the LB as usual.

5.2 Proactive Algorithm with the DSM Method (Pro&DSM)

When the users' distribution to the APs by the Pro is over, the Pro calls the DSMi or the DSMf to continue the LB and reduce the APs inter-communications as well.

5.3 Proactive Algorithm with Transfer and the DSM (Pro&transfer&DSM)

When the Pro has been achieved, different algorithms can follow it with intent to enhance the LB and reduce the APs inter-communications. The TAA can be applied followed by the DSMi/DSMf. This combination is the so-called *Pro&after&DSMi/DSMf*. Conversely, the TBA is applied before triggering the DSMi, i.e, the *Pro&before&DSMi*. However, in case the TBA is applied before triggering the DSMf, it is so-called the *Pro&before_1st&DSMf*. The TBA can be also applied after the replacement stage and then, it is called the *Pro&before_2nd&DSMf*. In the

first case, the priority is given to offload the small cells more than reducing the APs inter-communications as in the second case. In other words, the replacement stage of the DSMf is achieved as a first step in the Pro&before_2nd&DSMf and Pro&after&DSMf. Because, we observed that this significantly reduces the APs inter-communications. Otherwise, the early user transfer or handover processes will result in the loss of the users that can be served later on as BCs for replacement.

6 Performance Evaluation

6.1 Simulation Environments

The simulation is performed with a heterogeneous network with macro and small cells using *ns-3*. The proposed scenario consists of three macrocells and 10 small cells. Each set of three-hexagonal intersecting small cells forms a cluster. The user density threshold, D_{th} is equal to 18 users per cluster [7] [9]. The users allocate multi-traffic. Each user selects a specific bit rate in the range of 0 to 350 Mbps [7] [9] [13]. We consider a uniform deployment of small cells to diagnose the impact of the proposed algorithms on the network. Regarding the users' distribution, 50% of the mobile users were randomly distributed over the whole area, and the rest were fixed and uniformly distributed over the border areas, because the proposed algorithms aim to hand over the users located in the overlapping zones. The randomly distributed users follow the Circular Way (CW) mobility model [6], [14]. The users move in a circular path with a 10m radius and a speed of 3.6 km/h. The bandwidth for each small cell was set to 20 MHz. The transmission power for the small and macro cells was set to 24 dBm and 46 dBm, respectively. To model the path loss, we considered Non-Line-of-Sight (NLoS) propagation loss model [6]. To allocate the *PRBs* among the users in a cell, a Channel QoS-Aware (CQA) scheduler was adopted [6].

6.2 Performance Evaluation Metrics

To evaluate the performance, we considered four aspects: the load distribution across the small cells of the clusters, the Balance Improvement Ratio (BIR), the Balance Efficiency (BE) and the Reducing Inter-Communications Ratio between the APs (RICR). To measure the load distribution, the standard deviation (σ) and the Jain's fairness index (β) are considered. The BIR is expressed as done in [7] [9], as follows

$$BIR = \left| \frac{\sigma_{final} - \sigma_{initial}}{\sigma_{initial}} \right| \quad \textbf{(12)}$$

where $\sigma_{initial}$ and σ_{final} are the standard deviation across the small cells of the cluster before and after applying the LBA, respectively. We also took into account the signaling load, which is the summation of the handover rate, HOR for the Rea, the probability of rejection, PR for the users rejected from the macrocells, the transfer rate, TR for the transfer algorithms and the replacement rate, RR for the DSMf. The

BE is measured by considering the standard deviation and the signaling load, as done in [7] [9]. When applying the Rea, the DSMi or the Pro&DSMi, the BE is given by

$$BE_{rea/DSMi/Pro\&DSMi} = 1/(\sigma_{final} \times HOR) \quad \textbf{(13)}$$

Considering the DSMf or the Pro&DSMf, the BE is expressed as

$$BE_{DSMf/Pro\&DSMf} = 1/(\sigma_{final} \times (HOR + 2 \times RR)) \quad \textbf{(14)}$$

When the transfer algorithm with the DSMi/DSMf is considered, the BE of the Transfer&DSMi/DSMf and the Pro&transfer&DSMi/DSMf are given respectively by

$$BE_{transfer\&DSMi/Pro\&transfer\&DSMi} = 1/(\sigma_{final} \times (HOR + TR + PR)) \quad \textbf{(15)}$$

$$BE_{transfer\&DSMf/Pro\&transfer\&DSMf} = 1/(\sigma_{final} \times (HOR + TR + PR + 2 \times RR)) \quad \textbf{(16)}$$

Finally, the RICR% is expressed as done in [7] [9], as follows,

$$RICR\ \% = \left| \frac{\tau_{final} - \tau_{initial}}{\tau_{initial}} \right| \quad \textbf{(17)}$$

where $\tau_{initial}$ and τ_{final} are the load index before and after applying the DSMA.

6.3 Results Analysis

In the following, the results of the proactive algorithm with the transfer and DSM method is compared to the previous reactive and transfer algorithms [7] [9]. Fig. 4 shows that the Pro&before&DSMi achieves the best load distribution. We noticed that this algorithm often reaches the balance with the help of the Rea. Moreover, it outperforms the Pro&before_2nd&DSMf only by 2.04%. Alternatively, the worst load distribution is performed by the DSMf, as this latter focuses on reducing the APs inter-communications more than the LB. In total, the Pro&transfer&DSM is better than the Pro&DSM, the Transfer&DSM and the DSMA by 6.17%, 4.86% and 44.88%. Thus, it is recommended for the Pro to be followed by the transfer algorithms and DSMAs. Similar load distribution results are observed based on the β index.

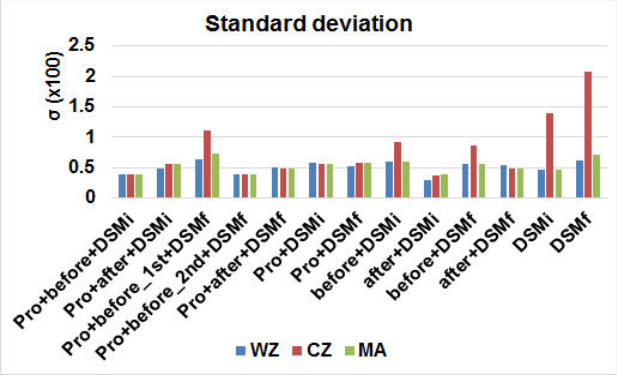

Fig. 4. Standard deviation for all the algorithms.

With regard to the BIR, the Transfer&DSM clarifies the best BIR. Because the Pro&transfer&DSM have already distributed the load well and then, there is no need to improve it more. To determine the best LBA, the signaling load is considered. In this context, we found that the Pro&transfer&DSMf requires the highest signaling

load due to the replacements. Note that each replacement requires two handovers. Conversely, the Pro&DSMi shows the lowest signaling load, as it only achieves handovers. In contrast, the Pro&transfer&DSMf requires all sorts of signaling. It is important to note that the Pro&before&DSMi rejects users from the BSs more than the Pro&before_2nd&DSMf by 50%. Regarding the BE, the Pro&DSMi achieves the best BE, as depicted in Fig. 5. In total, the Pro&transfer&DSM achieves a BE better by 1.85% and 28.16% than the Pro&DSM and the DSMAs, respectively. Besides, the BE of the Pro&transfer&DSMi is significantly better the Pro&transfer&DSMf.

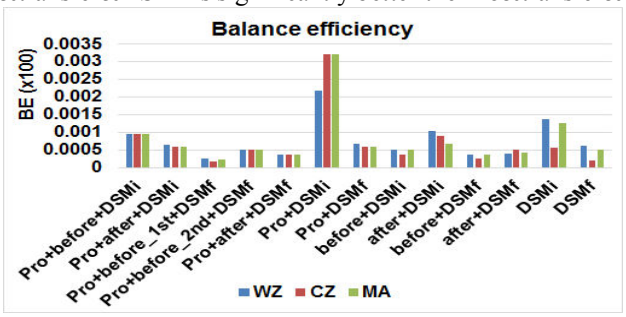

Fig. 5. Balance efficiency for all the algorithms.

Regarding the RICR, Fig. 6 shows that the DSMf with any algorithm reduces the APs inter-communications. The best RICR is achieved using the Pro&before_2nd&DSMf, Pro&after&DSMf and Pro&DSMf, which reaches 60.60%. In fact, since the Pro does not reject any user, this increases the probability of finding the BCs for replacement.

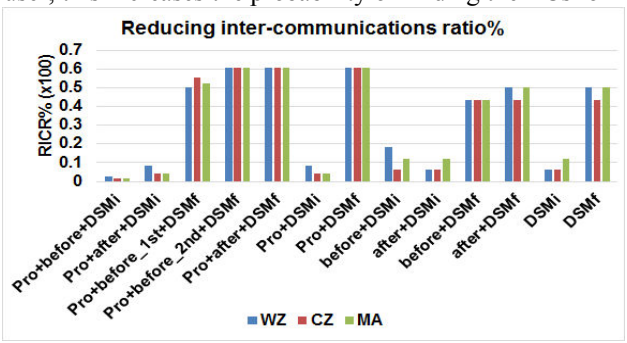

Fig. 6. RICR% for all the algorithms.

7 Conclusion

In this paper, several algorithms are proposed to load balance in UDNs. A proactive algorithm distributes the new users to the APs. The transfer algorithms offload the small cells and transfer the extra users to the macrocells before or after balancing the load by the reactive algorithm. To balance the load and reduce the inter-communications (RICR) between the APs as the same time, the design structure matrix (DSM) method is suggested. In this context, we found that although the

Pro&before&DSMi shows the best load distribution, the Pro&before_2nd&DSMf is better than it regarding the RICR. And, the Pro&DSMi outperforms it concerning the balance efficiency (BE). As a result, two promoting solutions would be recommended for load balancing in UDNs. If the BE has higher priority, the Pro&DSMi using the mixed algorithm would be an adequate solution. Instead, if the RICR is more important, the Pro&before_2nd&DSMf using the worst zone algorithm would be a promoting solution, as it significantly distributes the load and with a RICR of 60.60%.

References

1. X. Ge, S. Tu, G. Mao, C. Wang, and T. Han, "5g ultra-dense cellular networks," IEEE Wireless Commun., vol. 23, no. 1, pp. 72–79, Feb 2016.
2. Browning R. Tyson, "Applying the Design Structure Matrix to System Decomposition and Integration Problems: A review and new directions," IEEE Transactions on Engineering Management, 48:292–306, 2001.
3. Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Self-Configuring and Self-Optimizing Network (SON) Use Cases and Solutions, document TS 36.902, 3rd Generation Partnership Project, Sep. 2010.
4. S. Feng and E. Seidel, "Self-organizing networks (SON) in 3GPP long term evolution," Newsletter, Nomor Research GmbH, Munich, Germany, Tech. Rep., May 2008.
5. Z. Huang, J. Liu, Q. Shen, J. Wu, and X. Gan, "A threshold-based multi-traffic load balance mechanism in LTE-A networks," in Proc. IEEE Wireless Commun. Netw. Conf. (WCNC), pp. 1273–1278, Mar. 2015.
6. M. M. Hasan, S. Kwon, and J. H. Na, "Adaptive mobility load balancing algorithm for LTE small-cell networks," IEEE Trans. Wireless Commun., vol. 17, no. 4, pp. 2205–2217, Apr 2018.
7. M. Salhani and M. Liinaharja, "Load balancing algorithm within the small cells of heterogeneous UDN networks: Mathematical proofs," JOURNAL OF COMMUNICATIONS, vol. 13, no. 11, pp. 627-634, 2018.
8. I. Elgendi, K. S. Munasinghe and A. Jamalipour, "Traffic offloading for 5G: L-LTE or Wi-Fi," 2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), Atlanta, GA, pp. 748-753, 2017.
9. M. Salhani and M. Liinaharja, "Load Migration Mechanism in Ultra-Dense Networks". In Proceedings of the 2nd International Conference on Telecommunications and Communication Engineering (ICTCE 2018). ACM, New York, NY, USA, 268-274, 2018.
10. K. Dimou, M. Wang, Y. Yang, M. Kazmi, A. Larmo, J. Pettersson, W. Muller, and Y. Timner, "Handover within 3gpp lte: design principles and performance," in Proc. IEEE VTC, 2009.
11. M. Huang, S. Feng, and J. Chen, "A Practical Approach for Load Balancing in LTE Networks," Journal of Communications Vol. 9, No. 6, pp. 490-497 June 2014.
12. M. Salhani and M. Liinaharja. "Load Balancing in UDN Networks by Migration Mechanism with Respect to the D2D Communications and E2E Delay," JOURNAL OF COMMUNICATIONS, JCM14 (2019): 249-260, 2019.
13. P. Kela, Continuous Ultra-Dense Networks, A System Level Design for Urban Outdoor Deployments, book 1799-4942 (electronic), Aalto University publication series DOCTORAL DISSERTATIONS 86/2017.
14. C. Ley-Bosch, R. Medina-Sosa, I. A. González, and D. S. Rodríguez, "Implementing an IEEE802.15.7 physical layer simulation model with OMNET++," in Proc. 12th Int. Conf. Distrib. Comput. Artif. Intell., pp. 251–258, 2015.