Salhani, Mohamad

Comparison of the proactive and reactive algorithms for load balancing in UDN networks

Published in:
Journal of Communications

DOI:
10.12720/jcm.14.12.1119-1126

Published: 01/12/2019

Document Version
Publisher's PDF, also known as Version of record

Please cite the original version:
Salhani, M. (2019). Comparison of the proactive and reactive algorithms for load balancing in UDN networks. *Journal of Communications*, *14*(12), 1119-1126. https://doi.org/10.12720/jcm.14.12.1119-1126

Comparison of the Proactive and Reactive Algorithms for Load Balancing in UDN Networks

Mohamad Salhani
Aalto University, Espoo, 02150, Finland
Email: mohamad.salhani@aalto.fi

Abstract —Ultra-Dense Networks (UDNs) were introduced to support high data rate services and improve the network capacity. The load across the small cells is unevenly distributed owing to random deployment of small cells, the mobility of user equipments (UEs) and the preference of small cells during the selection/reselection. The unbalanced load causes performance degradation in both the throughput and successful handovers. Moreover, it may be responsible for radio link failures as well. To address this problem, this paper proposes different proactive algorithms to balance the load across UDN small cells and compare them to previous reactive algorithms. Proactive algorithms distribute the UEs, one by one, to the access points (APs), while the reactive ones are only triggered when the load of the chosen small-cell cluster reaches a predefined threshold. The numerical analysis shows that the load distribution achieved by the proactive algorithm with user rejection is better than that in the reactive algorithms by 34.97%. In addition, the impact of the small-cell cluster layout on the load balancing results is also studied in this paper. The results indicate that the load distribution and the balance improvement ratio in the intersecting small-cell model outperform those in the sequential small-cell one by 48.98% and 22.43%, respectively.

Index Terms—UDN, reactive algorithms, proactive algorithm with rejection, proactive algorithm without rejection, intersecting small-cell model, sequential small-cell model.

I. INTRODUCTION

To support the data demand for mobile broadband services and increase network capacity as well, the small cells will play an important role in the future 5G network and can significantly increase the capacity and throughput of the network [1], [2]. Due to the low cost of the small cells, subscribers may have their own small cells and deploy them anywhere, even to turn on and off at any time. Therefore, the small cells will be mostly randomly distributed throughout the network [3]. Since the small cells have low transmission power, only a few UEs can be served by each small cell, and the mobility of UEs leads to an unbalanced load across the network. In addition, the preference of small cells during cell selection and reselection loads more traffic onto them; this also causes an overloaded network. When UEs move onto overloaded small cells, the deficit in resources results in handover failures or poor quality of service (QoS). Hence, some small cells do not satisfy the QoS requirements, while other neighboring small cells resources remain unused.

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 [4]. The cSON has many features, like mobility robustness, optimization, mobility load balancing (MLB), interference management, and so on [5]. The MLB algorithm in a cSON optimizes the handover parameters and achieves load balancing (LB) without affecting the UE experience. Thus, it is necessary to study a load-balancing algorithm (LBA) that can adapt to various network environments and avoid the load ping-pongs.

II. RELATED WORK

Researchers have proposed several solutions to address the LB problem and enhance cellular network performance. The authors in [6] proposed an MLB algorithm considering constant-traffic UEs 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 [7], a traffic-variant UEs LBA has been proposed considering small cells; however, this algorithm also considered a fixed threshold to identify the overloaded cells. In [3], the authors proposed an MLB algorithm considering an adaptive threshold to decide overloaded cells in a small cell network. The algorithm estimates the loads in both overloaded cells and neighboring cells, and achieves handovers based on the measurements reported by UEs.

The authors in [8] mathematically proved the balance efficiency of the proposed LBAs based on the overlapping zones between the intersecting small cells. The authors focused on the optimization issue of the overlapping zone selection using different approaches. The proposed LBA was small cell cluster-based and aimed first to determine the best overlapping zone among several overlapping zones and then, to select the best UE for handover in order to reduce the number of the handovers and improve the performance of the whole UDN network. Nonetheless, the proposed algorithm was reactive, i.e., it is only executed when the user density of the chosen small-cell cluster reaches a predefined threshold.

Manuscript received March 25, 2019; revised November 8, 2019.
doi:10.12720/jcm.14.12.1119-1126

In this paper, we propose proactive algorithms that construct clusters of the small cells and perform the LB across the small cells. The proposed proactive algorithms are always on standby and ready to be triggered for distributing the new UEs to the small cells. For cluster formation, the algorithm considers an overloaded small cell and two neighboring small cells. Consequently, in each cluster, the algorithm performs the LB locally and updates cell individual offset (CIO) parameters of the cells. Simulation results show that the proposed proactive algorithm with rejection of the extra UEs improves the load distribution compared to the reactive algorithms proposed in [8]. Furthermore, this paper studies the impact of the small-cell cluster layout on the LB. The results indicate that the intersecting small-cell model proposed in [8] is better than the sequential small-cell one considered in this paper.

The rest of this paper is organized as follows: Section III describes the system model and assumptions we made. The different LBAs are proposed in Section IV followed by the performance evaluation in Section V. Section VI concludes the paper.

III. SYSTEM MODEL

A. System Description

We consider a heterogeneous LTE network composed of a set of macro cells and small cells, *N*, and a set of users, *U*, as done in [3], [8]. We consider the UDN small cells with overlapping zones and each set of small cells constitutes a so-called cluster. The LB is achieved in the small-cell clusters. In the simulation model, we considered a cluster consists of three intersecting small cells, which is called IC model, as done in [8], or three sequential small cells; SC model, as depicted in Fig. 1 (a) and (b), respectively. The purpose is to study the impact of the cluster layout on the LB results of the different LBAs.

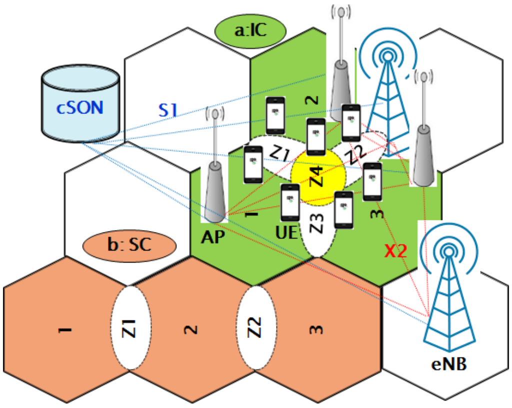

Fig. 1. System model with a cSON: IC model (a) and SC model (b).

The (small) cells interconnect with each other via X2 interface. This allows them to perform the needed functionalities such as handovers, load management, and so on [9]. Therefore, the UEs can move seamlessly among the cells. To optimize the parameters in the network, a cSON subsystem is considered [5]. The cells are connected to the cSON subsystem via S1 interface [10]. The cSON subsystem collects the required load-related information from the network and optimizes the parameters of the cells to perform the LB process.

B. Small Cells Load

To measure the small cells load in each cluster, the average resource block utilization ratio, *RBUR* is calculated from the physical resource blocks (*PRBs*) allocation information, as done in [3]. The small cell load, ρ_i, of cell *i* for a given time duration, *T*, is given as

$$\rho_i = \frac{1}{T.N_{PRB}} \sum_{j=1}^{u} RB_{(i,j)} \tag{1}$$

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

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

where *m* is the maximum number of the small cells constituting the cluster.

In order to determine overloaded, balanced and underloaded small cells in each cluster, we introduce two adaptive thresholds; upper and lower thresholds, δ_1, δ_2, respectively, which are defined, as done in [8] as follows

$$\delta_1 = ACL + \alpha \times ACL \tag{3}$$

$$\delta_2 = ACL - \alpha \times ACL \tag{4}$$

where α is the tolerance parameter, which controls the width of the balance zone. A small value of α requires many handovers to reach the needed LB, and vice-versa. In this paper, α is set to 0.05 [8]. Equation (3) and (4) show that the thresholds are a function of *ACL* and α.

C. Handover Procedure

In this paper, A3 and A4 event measurements are used to trigger a handover and select the UEs candidate for handovers, and the reference signal received power (RSRP) is assumed reporting signal quality for measurements, as done in [3], [11]. Actually, event A3 is widely used for triggering handovers in wireless networks [12]. In that way, event A3 is triggered and the UEs 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 [3]:

$$Mn + Ofn + Ocn - Hyst > Mp + Ofp + Ocp + Off \tag{5}$$

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 UEs in the overloaded cells to the underloaded cells. However, to balance the load, the system needs information about the edge-UEs distribution. For that, the event A4 is used. All the cells share the UEs information with the cSON. The condition for triggering event A4 is expressed as [3],

$$Mn + Ofn + Ocn - Hyst > Thresh \tag{6}$$

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

IV. PROPOSED LOAD BALANCING ALGORITHMS

In this following, we present the different LBAs that are proposed to balance the load across the small cells.

A. *Proactive Algorithm with (user) Rejection (ProR)*

The proactive algorithm with rejection (ProR) distributes the new UEs to the covering APs and rejects the extra users, as depicted in *Algorithm 1*. This algorithm is always on standby and ready to be triggered each time a new UE enters the network. For each new UE, the algorithm selects the best AP, which has the least load. In the ProR, the resources of the APs are considered limited; each AP has a maximum capacity, ρ_{th}. Therefore, when an AP is selected to include a new UE and the load of this AP, ρ_i will not exceed ρ_{th} if it accepts this UE, thus the UE is accepted. Otherwise, the ProR rejects the UE. This process is repeated for each new UE moves onto the network until the user density, D of the chosen cluster reaches the density threshold, D_{th}.

B. *Proactive Algorithm Without (user) Rejection (Pro)*

The proactive algorithm without rejection (Pro) is similar to the ProR, as depicted in *Algorithm 2*; however, the APs are considered having enough resources (e.g. ρ_{th} is greater than that in the case of ProR by 20%) to accept the new UEs as long as the user density of the current cluster does not exceed D_{th}. In practice, the density condition is not necessary to be checked, as this algorithm is always on standby and triggers for each new UE. This condition is only imposed in this study to compare the results of these two proactive algorithms to those in the reactive algorithms with the same user density.

C. *Reactive Algorithm (Rea)*

The reactive algorithm (Rea) has been proposed in [8] to balance the load across the APs in the IC model. Nevertheless, this algorithm is only triggered once the user density of the cluster reaches D_{th}. To achieve the reactive algorithm, the authors have suggested three approaches based on the overlapping zones concept. In the c*ommon zone (CZ) approach,* the load is only balanced via the UEs that are located in the CZ between the three overlapping small cells; zone 4 (Z_4), as shown in Fig. 1. In the SC model that is proposed in this paper, the CZ approach cannot be applied, since there is no CZ between all the three sequential small cells. The second approach is the so-called *worst zone (WZ) approach*. The LB in this approach is achieved in the WZ, which has the smallest value of the Jain's fairness index, β (explained later). Note that the balance efficiency of the WZ approach has been mathematically proven in [8]. The third approach is the *mixed approach (MA).* This approach is a hybrid approach that combines the CZ approach and the WZ approach. It starts balancing the load in the CZ and then, it transits into the WZ with or without returning to the CZ. Hence, in this paper we can only adopt the WZ approach in the SC model.

The reactive algorithm, which has been proposed in [8], is adopted again in this paper in order to compare it to the proactive algorithms. This algorithm is periodically executed in the cSON subsystem. To achieve the LB, the algorithm needs to identify the cluster with the highest density and then, the overlapping zone and the best candidate UE (BC) to be handed-over. For that, it **first** starts checking the user density, D within each cluster and then, it compares the density of the cluster with the highest density to the density threshold, D_{th}. If the user density does not exceed the threshold, the algorithm is stopped. Otherwise, the algorithm sets the UE's load, $RBUR_j$ of each UE$_j$, its zone and the tolerance parameter α. Next, the algorithm calculates the load of each AP, ρ_i, and the *ACL* with (1) and (2), respectively. Meanwhile, the algorithm determines the state of each AP by the transfer policy. This policy verifies which AP must exclude an UE (overloaded AP) and which one must include this UE (underloaded AP). For that, two thresholds, δ_1 and δ_2 with (3) and (4) are needed. According to the transfer policy, an underloaded AP can accept new UEs and handed-over UEs from an overloaded AP. A balanced AP can only accept new UEs, while an overloaded AP does not receive any new or handed-over UEs. 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 calculates the Jain's fairness index (β) [13] as

$$\beta = {\left(\sum_{i=1}^{n} \rho_i\right)^2} \Big/ {\left(n \times \left(\sum_{i=1}^{n} {\rho_i}^2\right)\right)} \tag{7}$$

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. When all the APs have the same load, β is equal to one. Otherwise, β approaches 1/n, so $\beta \in [1/n, 1]$. The **third step** is to apply the selection policy for identifying the BC to be handed-over. For that, the difference (Δ)

between the load of the chosen overloaded AP and the *ACL* is calculated by

$$\Delta = \rho_{overloaded\_AP} - ACL \tag{8}$$

Of all the UEs 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 UE's load and Δ has the smallest absolute value as follows

$$BC_j = \left|RBUR_j - \Delta\right| \tag{9}$$

The **fourth step** is to calculate the new β if the BC is handed-over. This is performed by the distribution policy to ensure that the expected handover will definitely improve the balance before achieving the handover. Thus, the handover will be carried out if and only if β_{new} is greater than β_{old}. If this condition is satisfied, the algorithm selects this BC and the handover occurs. 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 selected LB scheme. For instance, the target zone in the WZ approach is the zone that has the smallest value of β, as depicted in *Algorithm 3*. Then, the algorithm repeats the last policies in the new target zone. The **fifth step** is to check again if there is still an overloaded AP, and also if the balance improvement is still valid. If so, the LB enhancement is evaluated in the new target zone and so on. Otherwise, the algorithm is stopped and waits for the next trigger.

D. Shifting Algorithm (SA)

In the SC model, we found that the WZ algorithm (WZA) demonstrates unsatisfied LB results and shows its limitation. Actually, the WZA is unable to balance the load in the scenarios in which a balanced AP is located between two overloaded APs or is located between an overloaded AP and an underloaded AP. Other scenarios can be considered in which an overloaded AP is located between an overloaded AP and a balanced or an underloaded AP. These four cases require shifting (handing over) the UEs and these cases are the so-called "shift conditions". In contrast, the WZA slightly improves the LB in case the underloaded AP is located between an overloaded AP and an underloaded or a balanced AP. In these two last cases, the LB will be exclusively between only two APs. To overcome this limitation, the *shift algorithm (SA)* is proposed, as illustrated in *Algorithm 4*. The SA is composed of the shifting stage and the balancing stage that is achieved by the ordinary WZA. The **first step** and the **second step** of the SA are the same as the WZA. The **third step** is to check the shift conditions, i.e., the chosen cluster is one of the four cases that require shifting. If these conditions are not satisfied, the WZA is executed as usual. Otherwise, the **fourth step** is to check the possibility of applying the WZA for only one handover. If this handover is not achievable, the SA definitely converts into the WZA. Otherwise, the shifting stage starts by calculating Δ_{shift} as the difference of the load of the most loaded AP, ρ_{ml} and the next loaded AP, ρ_{nl} as follows,

$$\Delta_{shift} = \rho_{ml} - \rho_{nl} \tag{10}$$

To make the shifting decision, the **fifth step** is to check if the Δ_{shift} is positive, i.e., the AP, which is located on the sides (e.g. AP_3 in Fig. 1), is still the most overloaded AP. If so, an UE should be shifted from the most overloaded AP (AP_3) to the least overloaded one (AP_2) (or to the balanced AP in other cases), even though the latter became balanced after the first step of the WZA. The best UE, that can be shifted, is the one for which the difference of its load and Δ_{shift} has the smallest absolute value. Note that the shifted UEs cannot be handed-over again with the underloaded AP (AP_1) during the balancing stage, as these UEs are not located in Z_1, which is the overlapping zone between AP_1 and AP_2. For this reason, the SA achieves many handovers to reach the required balance. Furthermore, during the shifting stage, the distribution condition does not need to be checked. After that, the algorithm repeats the handover procedure with another UE using the balancing stage, if possible, and so on. This process is repeated as long as Δ_{shift} is positive and the AP in question is still overloaded. Otherwise, the SA definitely converts into the WZA.

Algorithm 1: Proactive algorithm with rejection (ProR)

```
1: Get RSRP and PRB measurements of UE j and cell i, $D_{th}$ and UE's zone
2: if $D < D_{th}$ then
3:   Find the cell that covers this UE and has the smallest $\rho_i$
4:   if $\rho_i < \delta_1$ and $(\rho_i + RBUR_j) > \rho_{th}$ then
5:     Reject this UE and update the call drop rate (PR)
6:   else
7:     Transfer the new UE to the target cell
8:     Update $\rho_i$ of the target cell
9:   end if
10: end if
```

Algorithm 2: Proactive algorithm without rejection (Pro)

```
1: Get RSRP and PRB measurements of UE j and cell i, $D_{th}$, and UE's zone,
2: if $D < D_{th}$ then
3:   Find the cell that covers this UE and has the smallest $\rho_i$
4:   Transfer the new UE to the target cell
5:   Update $\rho_i$ of the target cell
6: end if
```

Algorithm 3: Worst zone algorithm (WZA)

```
1: Get RSRP and PRB measurements of UE j and cell i, $D_{th}$, UE's zone and $\alpha$
2: Find the cluster with the highest user density
3: if $D >= D_{th}$ then
4:   Calculate $\rho$ for each cell i, ACL, $\delta_1$ and $\delta_2$
5:     if one of the chosen cluster's cell has $\rho_i > \delta_1$ then
6:       Calculate $\beta_1$, $\beta_2$, $\beta_3$ and $\beta_4$, and then find the worst zone
7:       Apply the transfer policy
8:       Calculate $\Delta$ and determine the BCj
9:       if $\beta_{new} > \beta_{old}$ then
10:        Transfer the BCj to the target cell (execute a handover)
11:        Update $\rho$ for each cell i and go to step 5
12:      else
13:        if there are UEs of 2nd order then
14:          Find the new BCj and execute a handover
15:          Update $\rho$ for each cell i and go to step 5
16:        else
17:          Transfer to the zone of 2nd order and go to step 7
```

```
18:          end if
19:       end if
20:    else
21:       if there is a cluster of the next order then
22:          Go to step 3
23:       end if
24:    end if
25: end if
```

Algorithm 4: Shift algorithm (SA)

```
1: Get RSRP and PRB measurements of UE j and cell i, $D_{th}$, UE's zone and $\alpha$
2: Find the cluster with the highest user density
3: if $D >= D_{th}$ then
4:    Calculate $\rho$ of each cell i, ACL, $\delta_1$ and $\delta_2$
5.    if one of the chosen cluster's cell has $\rho_i > \delta_1$ then
6:       Calculate $\beta_1$ and $\beta_2$, and then find the worst zone
7:       if the shift conditions are met then
8:          if one handover is executable then
9:             Execute a HO by WZA
10:         end if
11:            Calculate $\Delta_{shift}$
12:            if $\Delta_{shift} > 0$ then
13:               if $\rho_{i(on\ side)} > \delta_1$ then
14:                  if an UE can be shifted then
15:                     Execute one shift and then, go to step 8
16:                  else
17:                     Go to step 8
18:                  end if
19:               else
20:                  Go to step 8
21:               end if
22:            else
23:               if $\rho_{i(on\ side)} > \delta_1$ then
24:                  Go to step 8
25:               else
26:                  Apply the WZA policies
27:               end if
28:            end if
29:      else
30:         Apply the WZA policies
31:      end if
32:   else
33:      Find the cluster of the next order and go to step 3
34:   end if
35: end if
```

V. PERFORMANCE EVALUATION

A. Simulation Environments

In order to evaluate the performance of the proposed algorithms and compare their results to the previous reactive algorithms, we performed the simulation with a heterogeneous network with macro and small cells. The proposed scenario consists of three macro cells and 10 small cells. Each set of three-hexagonal intersecting small cells (IC model) or sequential small cells (SC model) forms a cluster. The user density, D is on average equal to six UEs per small cell. Therefore, the density threshold, D_{th} is equal to 18 UEs per cluster, as considered in [8]. The UEs allocate multi-traffic. Each UE selects a specific bit rate in the range of 0 to 350 Mbps [8], [14].

We consider a uniform deployment of small cells in order to diagnose the impact of the proposed algorithms on the network from different aspects. With regard to the UEs distribution, 50% of the mobile UEs were randomly distributed over the whole area, and the rest were fixed and uniformly distributed over the border areas of the small cells, because the proposed algorithms aim to hand over the UEs located in the overlapping zones. The randomly distributed UEs follow the circular way (CW) mobility model [3], [15]. In this mobility model, the UEs 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 cells 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 [3], [16]. To allocate the *PRBs* among the UEs in a cell, a channel QoS-aware (CQA) scheduler was adopted [3], [17]. More parameters are listed in Table I.

TABLE I: SIMULATION PARAMETERS

Parameters	**Values**
Number of small cells	10
Tx power	24 dBm (small cell) and 46 dBm (macro cell)
System bandwidth	20 MHz
Antenna mode	Isotropic
Pathloss	$PL=147.4+43.3\log_{10}(R)$
Fading	Standard deviation 4 dB, lognormal
Resource scheduling	CQA scheduler
CIO_{min} and CIO_{max}	-6dB, 6dB
Hysteresis	2 dB
ρ_{th}	1Gbps
D_{th}	18 *UE*
UE velocity	3.6 km/h
Mobility model	Uniform, 50% CW mobility UEs and 50% static UEs

B. Performance Evaluation Metrics

To evaluate the performance, we considered three aspects: the load distribution across the small cells, the balance improvement ratio (*BIR*) and the balance efficiency (*BE*). To measure the load distribution, the standard deviation (σ) and the Jain's fairness index (β) with (7) are considered. The *BIR* is expressed as done in [8],

$$BIR = \left|\frac{\sigma_{final} - \sigma_{initial}}{\sigma_{initial}}\right| \tag{11}$$

where $\sigma_{initial}$ and σ_{final} are the standard deviation of the loads among the small cells of the cluster before and after applying the LBA in question, respectively.

We also took into account the signaling load, i.e., the handover rate, *HOR* for the reactive algorithms, and the probability of rejection (call drop rate) of the new incoming UEs, *PR* for the ProR.

The *BE* is measured by considering the standard deviation and also the signaling load performed in each algorithm, as done in [8]. When applying the reactive algorithm, the *BE* is given by

$$BE_{rea} = {}^{1}\!/\!(\sigma_{final} \times HOR) \tag{12}$$

By applying the ProR or the Pro, the *BE* is expressed respectively as

$$BE_{ProR} = {}^{1}\!/\!(\sigma_{final} \times PR) \tag{13}$$

$$BE_{Pro} = {}^{1}\!/\!\sigma_{final} \tag{14}$$

C. *Results Analysis*

To analyze the results and evaluate the performance of the proposed algorithms, we compare the results of the proposed proactive algorithm with or without rejection to the previous reactive algorithms proposed in [8]. The comparison is accomplished for both small-cell cluster layouts; the IC model and the SC model.

Fig. 2 shows the standard deviation of the load distribution across the small cells of the cluster versus the running time, for the different algorithms. In the IC model, we notice that the ProR shows the smallest value of the standard deviation, while the Pro leads to the worst load distribution. In fact, the Pro distributes the new UEs similar to the ProR; however, the incoming UEs, which are not rejected when the Pro is applied, will deteriorate the LB process across the small cells. Furthermore, the ProR improves the load distribution compared to the reactive algorithm (the average value of σ for the CZ, WZ and MA algorithms) by 34.97%. Moreover, the worst algorithm among the reactive algorithms is the CZ algorithm, since only the UEs located in the CZ can be handed-over.

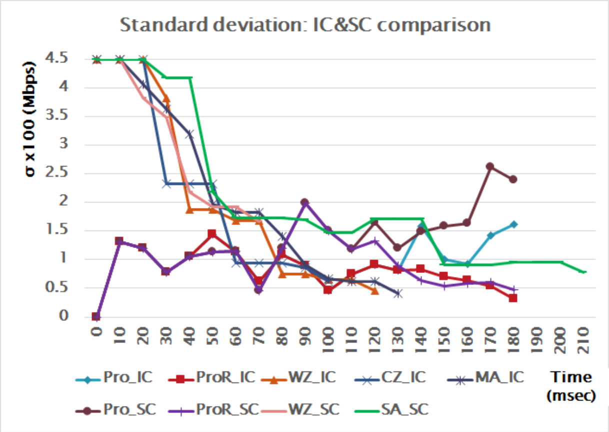

Fig. 2. A comparison of the different σs for the considered algorithms.

In the SC model, the load distribution achieved by the ProR is also better than the Pro, WZA and SA. In addition, the load distribution performed by the SA is better than the WZA by 53.19%. This is at the price of higher running time and the complexity of the SA, which requires more processing time due to the frequent calculations of deltas (Δ, Δ_{shift}). Furthermore, the WZA in the IC model achieves a load distribution better than the ProR in the SC model by 6.25%. In total, the load distribution in the IC model outperforms that in the SC model by 48.98%. Because there are four overlapping zones to select the BCs in the IC model against only two overlapping zones in the SC one. It is important to note that similar load distribution results are obtained based on the Jain's fairness index, β.

Actually, to compare the LB results of the SC model to those in the IC one, we noticed the following common metrics between these two models: β_1, β_2 and σ. The WZA can be applied in both models as well. The scenario of this comparison is simulated with 100 UEs, and the data traffic for each one of the UEs was set at a guaranteed bit rate (GBR) of 512Kbps. In this context, Fig. 3 clarifies that the SA takes more running time than the other algorithms, while the WZA in the IC model achieves the required balance faster than any other algorithm. This is because the SA starts shifting the UEs from AP$_3$ to AP$_2$ and then, it starts balancing the load. However, the WZA in the IC model can directly hand over the UEs from AP$_3$ to AP$_1$ in Z_3 that does not exist in the SC model. Accordingly, the index β_1_IC(WZ) is greater than β_1_SC(SA), and this latter is greater than β_1_SC(WZ). The same results are confirmed for β_2. Likewise, Fig. 3 clarifies that σ_IC(WZ) is smaller than σ_SC(SA) and this latter is smaller than σ_SC(WZ). Subsequently, the IC model distributes the load across the small cells better than the SC one.

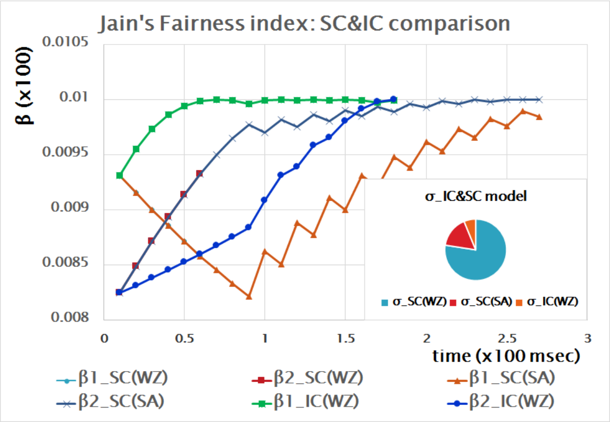

Fig. 3. Comparison of the different *βs* and the σs in the IC&SC models.

With regard to the *BIR* achieved by each algorithm, Fig. 4 demonstrates that the best *BIR* is carried out by using the reactive algorithms in the IC model (average *Rea*), which is better than that in the SC model by 22.43%. Furthermore, in the SC model, the *BIR* using the SA is better than that in the case of the WZA by 31.27%. Alternatively, the *BIR* using the WZA in the IC model is higher by 42.82% than that in the SC model.

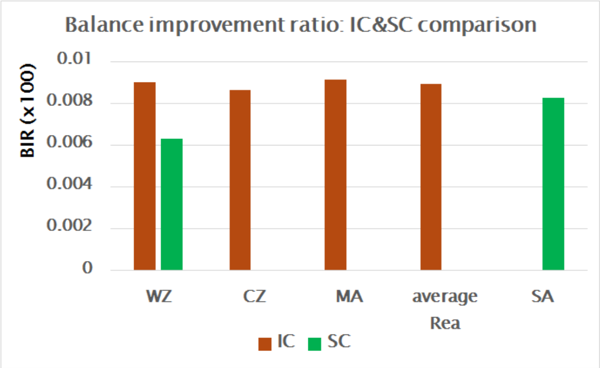

Fig. 4. *BIR* for the different algorithms in the IC&SC models.

In order to determine the best LBA, the signaling load caused by each algorithm is considered. Fig. 5 shows the *HOR* for the reactive algorithms and the *PR* for the ProR. We observe that the *HOR* in the IC model is higher only by 2.05% than that in the SC model at the expense of better load distribution in the IC model. Because the UEs located in Z_4 can be handed-over among three APs, not only between two APs like in the SC model. Moreover, the SA leads to the highest *HOR* due to many shifting processes needed to reach the required balance. Conversely, the *PR* in the IC model is higher than the *HOR* using the reactive algorithms by 35.67%. Additionally, the *PR* in the SC model outperforms that in the IC one by 5.55%, as the incoming UEs in the SC model can only be accepted by one of two APs.

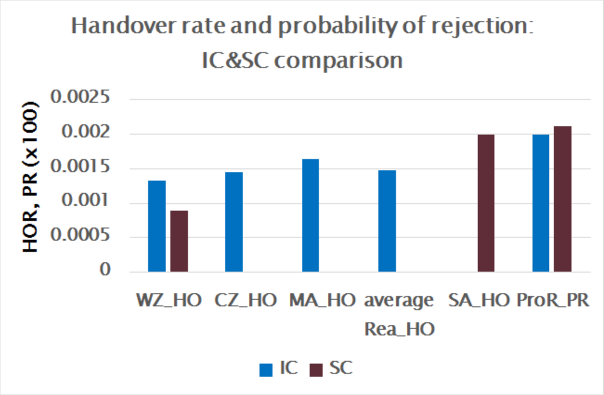

Fig. 5. *HOR* and *PR* for the different algorithms in the IC&SC models.

On the other hand, we found that the IC model significantly improves the *BE* compared to the SC one, as shown in Fig. 6. In the IC model, the BE_{ProR} is better than the BE_{Pro} and the BE_{rea} by 24.45% and 9.09%, respectively. On the contrary, the BE_{WZA} outperforms the BE_{ProR} only by 5.09%. The worst *BE* is noticed using the CZ algorithm. Nevertheless, the BE_{CZ} is still much better than the BE_{Pro}. Moreover, the MA algorithm clearly enhances the *BE*, but this algorithm results in a higher signaling load and requires more processing time as well. In the SC model, the BE_{ProR} is better than the BE_{Pro}, BE_{SA} and BE_{WZA}. Although the load distribution performed by the SA is better than that by the WZA; however, the BE_{WZA} outperforms the BE_{SA} by 5.30%. Besides, the load distribution outcomes and the *BE* in the IC model are much better than those in the SC one. As a result, to balance the load based on the small-cell cluster and the overlapping zones concept, the SC model is not preferred.

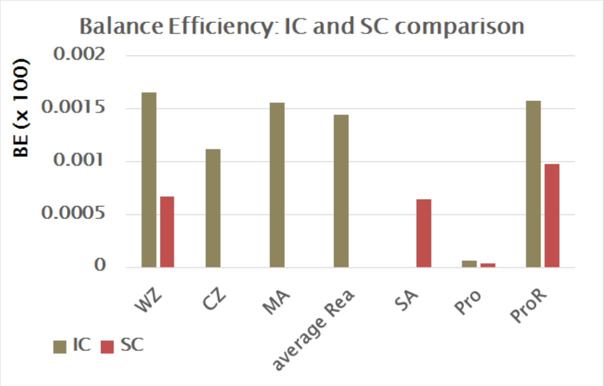

Fig. 6. *BE* for the different algorithms in the IC&SC models.

VI. CONCLUSION

In this paper, two proactive algorithms for balancing the load in UDN networks are proposed. The proactive algorithm with user rejection (ProR) distributes the new UEs to the APs and rejects the extra UEs that overload the target cells, while the proactive algorithm without user rejection (Pro) does not reject any extra UE and this leads to deteriorate the load balancing (LB). The proposed proactive algorithms are compared to the previous reactive algorithms; worst zone algorithm (WZA), common zone algorithm and the mixed algorithm. The impact of the small-cell cluster layout on the LB is also studied in this paper. The intersecting small-cell (IC) model is significantly better than the sequential small-cell (SC) one. As a result, to construct a cluster for balancing the load across the small cells based on the overlapping zones concept, two choices are possible: a WZA or a ProR. Although the WZA shows the best balance efficiency (*BE*) with a handover rate of 13.33%, the ProR achieves the best load distribution with a call drop rate of 20%. The *BE* of the WZA is only better by 5.09% than that of the ProR. Future works will deal with the LB using the design structure matrix (DSM) method, which can be used to reduce the end-to-end delay for the users communicating within UDN networks, and to balance the load as well.

REFERENCES

[1] J. Hoadley and P. Maveddat, "Enabling small cell deployment with HetNet," *IEEE Wireless Commun.*, vol. 19, no. 2, pp. 4–5, Apr. 2012.

[2] Qualcomm: The 1000x Data Challenge. Accessed: Sep. 10, 2016. [Online]. Available: https://www.qualcomm.com/invention/1000x

[3] 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.

[4] 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.

[5] S. Feng and E. Seidel, "Self-organizing networks (SON) in 3GPP long term evolution," *Newsletter, Nomor Research GmbH, Munich, Germany, Tech. Rep.*, May 2008.

[6] N. Zia and A. Mitschele-Thiel, "Self-organized neighborhood mobility load balancing for LTE networks," in *Proc. IFIP WD*, Nov. 2013.

[7] 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)*, Mar. 2015, pp. 1273–1278.

[8] 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.

[9] Evolved Universal Terrestrial Radio Access Network (E-UTRAN); X2 Application Protocol (X2AP), document TS 36.423, 3rd Generation Partnership Project, Sep. 2014.

[10] "Evolved universal terrestrial radio access network (E-UTRAN); S1 application protocol (S1AP)," 3rd Generation Partnership Project (3GPP), TS 36.413.

[11] "Evolved universal terrestrial radio access (E-UTRA); radio resource control (RRC); protocol specification," 3rd Generation Partnership Project (3GPP), TS 36.331.

[12] 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.

[13] 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.

[14] 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.

[15] 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.*, 2015, pp. 251–258.

[16] J. B. Andersen, T. S. Rappaport, and S. Yoshida, "Propagation measurements and models for wireless communications channels," *IEEE Commun. Mag.*, vol. 33, no. 1, pp. 42–49, Jan. 1995.

[17] J. M. Ruiz-Avilés, *et al*., "Design of a computationally efficient dynamic system-level simulator for enterprise LTE femtocell scenarios," *J. Electr. Comput. Eng.*, vol. 2012, Oct. 2012.

Mohamad Salhani is an associate professor at the Department of Computer and Automation Engineering (CAE), Faculty of Mechanical and Electrical Engineering (FMEE), Damascus University since 2016. He received his B.S degree in Electrical Engineering. from the FMEE in 2000, M.Sc degree from National Polytechnic Institute of Lorain (INPL), France in 2005 and Ph.D degree from National Polytechnic Institute of Toulouse (INPT), France in 2008. He was an assistant professor at the CAE, FMEE, at Damascus University in 2009. In 2016, he was a vice-dean for Administrative and Scientific Affairs at the Applied Faculty, Damascus University. He is currently a visiting professor at the Department of Communications and Networking, School of Electrical Engineering, Aalto University, Espoo, Finland. His research interests include 5G mobile communication systems, Ultra-dense networks (UDNs), Internet of Things and LoRa technology.