A Load Balance Mechanism in Heterogeneous Network Based on Utility Function

Qing Wang[1], Minjiong Zhu[1], Yongrun Yu[1], Maode Ma[2], and Xin Cui[1]
[1] School of Electronic Information Engineering, Tianjin University, Tianjin, China
[2] School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798, Singapore
Email: wangq@tju.edu.cn; zhuminjiong@sina.cn; 498699845@qq.com; EMDMa@ntu.edu.sg; 543657726@qq.com

Abstract —The coexistence of LTE, WiFi and WiMax forms the heterogeneous network (HetNet) nowadays. The most common problem in HetNet is the load balance between different types of network, which would be even more difficult when the users' demand is considered. For the purpose of this paper, we proposed a load balance algorithm based on utility function (LBUF). This algorithm considers network load level and users' demand at the same time. A self-adaptive threshold is defined for dynamically controlling the load between the networks according to the number of loads in the whole network. What's more, the LBUF algorithm needs to gather the global information of the HetNet, and the software-defined network (SDN) is able to provide the data we need. So the LBUF algorithm is suggested being installed in the SDN controller. The numerical simulation results demonstrated the fast convergence of our approaches.

Index Terms—Load balance, utility function, software-defined network, heterogeneous network

I. Introduction

With the development of wireless communication technologies, HetNet consisting of LTE, WiFi and WiMax gradually appears. Each network has its own advantages so that users have the chance to know and choose them. Therefore, different types of network will certainly coexist in the next generation network. Accordingly, intelligent mobile terminals are already equipped with multi wireless interface in order to access different network. Since the user equipment may independently choose one network, the load in the HetNet is very likely to become unbalanced, which may cause the low radio resources utility rate and bad user experience. To avoid unbalanced load distribution, a proper load balance method is needed. However, load balance is a challenging problem because not only the load distribution in the HetNet, but also network characteristics (such as data rate, link quality and so on) should be considered at the same time to make an optimized decision. In addition, users' experience is becoming increasingly important in terms of evaluating the performance of a network, so the preference of user should be considered in the load balance process.

Many algorithms have been proposed to make the loads distribute more evenly in the whole network. [1] provided an iteration method based on signal strength to adjust the handover offset, which will avoid ping-pang and short time stay effect. [2] suggested a utility-oriented load balance algorithm. They offload the traffic to multi light-loaded networks to avoid ping-pang and hidden cell effect. [3] proposed a load balance method based on fuzzy logic and utility functions concerning bandwidth, block rate and average transfer time. Although their algorithm has good performance, they don't take comprehensive network characteristics into consideration. What's more, the preference of user is not demonstrated either. [4] provided a load balance method that considered the network characteristics and users' demand at the same time, but they lack the precise proof of the convergence of the algorithm and their utility function still has the probability to be improved. What's more, they used a fixed threshold value to judge whether the load balance process should be triggered or not, which is lacking in flexibility. In this paper, we will set a self-adaptive threshold and give a proof of the convergence of the load balance algorithm.

In order to avoid simply evaluating one network by single criterion, carefully designed utility functions are used to describe the utility of the network. [5]-[7] suggested abundant kinds of utility functions. In [5], the form of sigmoid function is widely used and the parameters in the function are adjustable, which means one utility function can be used in different cases by changing its parameters. But [6] and [7] pointed out that the traditional additive aggregation method has limitations when one of the utility values that are used in the aggregation expression is close to zero. They proposed a multiplicative method to overcome the limit of the additive aggregation. Their approach is attractive and we plan to use the multiplicative method for utility calculation. A proof is given in this paper to confirm the ideal effect of this multiplicative method.

As the load balance algorithm based on utility function (LBUF) requires the global data of the whole network that the traditional network cannot provide, our algorithm is suggested being installed in the Software-Defined Network (SDN) controller. The booming SDN

Manuscript received June 25, 2016; revised September 20, 2016.
This work was supported by the 863 program under Grant 2015AA01A706.
Corresponding author email: EMDMa@ntu.edu.sg
doi:10.12720/jcm.11.9.871-878

technology separates the data plane and the control plane, and at the same time provides us with the global view of the whole network [8]. The data that LBUF algorithm needs can be obtained through the south-bound interface from network infrastructure and the command that need to be implemented is transported through the south-interface, too. With global data, the LBUF algorithm will quickly judge the load distribution in the HetNet and make the correct balance decision.

In this paper, we have three innovation points as following:

(1) We applied the sigmoid function, which was applied to a train communication network [5], to a load balance algorithm;

(2) We proved the convergence of the algorithm;

(3) We proposed a self-adaptive threshold to judge the status of the HetNet.

The rest of this paper is organized as below: the system model is shown in the second section. The third section is responsible for illustrating the LBUF algorithm in detail, which includes the elaboration of utility functions, the convergence of load balance algorithm and the corresponding pseudo-code. At last a numerical simulation is made to confirm that LBUF algorithm has satisfied performance.

II. System Model

The scenario considered in this paper is shown in Fig. 1. The SDN architecture is used to gather a global view of the HetNet and make the configuration of the algorithm easier. On the one hand, the global view of the HetNet is derived from the southbound interface(i.e., each network reports its status through southbound interface). On the other hand, the LBUF algorithm is installed in the SDN controller and it is configured through the northbound interface. The LBUF algorithm calculates each network's utility and according to the utility value, the LBUF algorithm makes the load balance decision, which will be sent to the user equipment through the southbound interface.

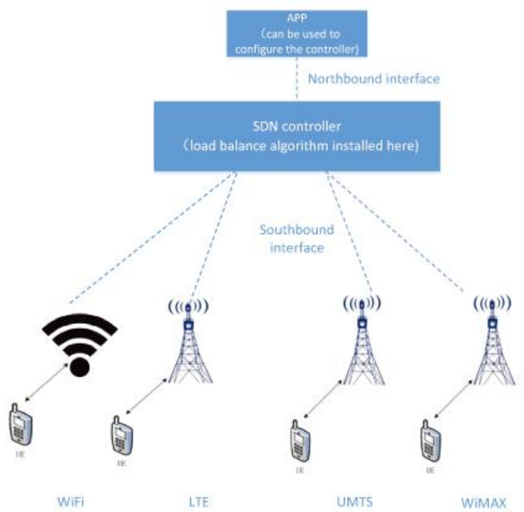

Fig. 1. Network scenario.

In this paper, network utility is calculated by multi-criteria utility function which concerns four network characteristics: data rate, power consumption, money cost and link quality. Every characteristic has its own single-criteria utility function and the multi-criteria utility function is responsible for combining the single-criteria utility functions in an appropriate way. Every user provides their preference to the SDN controller through southbound interface for the utility calculation. The LBUF algorithm calculates every network's utility value for each user and makes the load balance decision according to the load distribution in the HetNet.

III. System Model

A. Utility Functions used in LBUF Algorithm

in this part, we will discuss the single-criteria utility function and multi-criteria utility function for network selection. for the purpose of single-criteria utility function, we mainly focus on the math characteristics such as domain of definition, range, concavity and convexity. for the purpose of multi-criteria utility function, we mainly concentrate on how to aggregate several single-criteria utility functions in an appropriate way.

1) Single-criteria utility function: Single-criteria utility function is responsible for calculating the utility of one network characteristic and the utility value is the bigger the better. As the value of utility eventually decides the network selection, the utility function should be carefully designed. For the purpose of this paper, we use utility function proposed in [6]. The specific expression of the function is showed below:

$$u(x)=\begin{cases} 0, & x<x_\alpha \\ \dfrac{\left(\dfrac{x-x_\alpha}{x_m-x_\alpha}\right)^\zeta}{1+\left(\dfrac{x-x_\alpha}{x_m-x_\alpha}\right)^\zeta}, & x_\alpha<x<x_m \\ \dfrac{\left(\dfrac{x-x_m}{x_\beta-x_m}\right)^\gamma}{1+\left(\dfrac{x-x_m}{x_\beta-x_m}\right)^\gamma}, & x_m<x<x_\beta \\ 1, & x>x_\beta \end{cases} \tag{1}$$

where

$$\gamma=\frac{\zeta\left(x_\beta-x_m\right)}{x_m-x_\alpha} \tag{2}$$

$$\zeta\geq\max\left\{\frac{2\left(x_\beta-x_m\right)}{x_m-x_\alpha},2\right\} \tag{3}$$

In this formula, variable x is one parameter that a network offers. The value of $u(x)$ represents the utility of parameter x and the range of $u(x)$ is deliberately scaled to [0,1]. x_α and x_β is the lower bound and upper bound of variable x respectively. x_m is the inflection point of the function. Parameters γ and ζ are responsible for controlling the sharpness of the function.

An image of a single-criteria utility function is shown in Fig. 2.

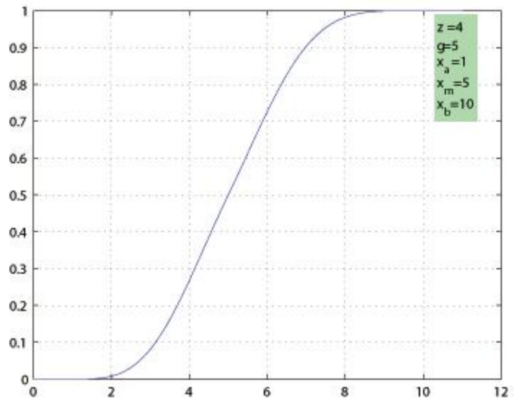

Fig. 2. Single-criteria utility function.

Particularly, the expression of $u(x)$ above is feasible when x is a positive factor like data rate and link quality, the value of which are the bigger the better. Of course, there exist negative factors such as cost and power consumption which are the smaller the better. The single-criteria utility function of negative factors has also been proposed in [6] as $v(x)=1-u(x)$.

2) Multi-criteria utility function: Multi-criteria utility function is responsible for combining single-criteria utility functions according to user preference weight. We model user preference by assigning weight to each utility value. We denote the set of weight of user preference as $\{\omega_i\}$, where $\sum_{i=1}^{n}\omega_i=1$, n is the total number of the existing networks. The value of multi-criteria utility function finally determines which network should be selected: just choose the network with the maximum value.

Traditional multi-criteria utility function is the additive aggregation. The specific expression of additive method is as below:

$$U(\mathrm{X})=\sum_{i=1}^{n}\omega_i u_i(x) \tag{4}$$

But [7] points out that such model has obvious limit and may result in making wrong network selection decision and they proposed another aggregation function as below:

$$U(\mathrm{X})=\prod_{i=1}^{n}[u_i(x)]^{\omega_i} \tag{5}$$

They have proved some characteristics of this function. What they did not mention is the effect of ω_i. As the multiplicative form is relatively unusual, it is necessary to interpret that ω_i which is placed at the exponent position can really represent the user preference. For example, assume that $u_1(x_1)$ is the minimum value of the set $\{u_i(x_i)\}$ and ω_1 is the weight assigned to $u_1(x_1)$. Logically, in this circumstance, if we increase ω_1, which means the user attaches more importance to utility value $u_1(x_1)$, the value of multi-criteria utility function should be smaller because $u_1(x_1)$ is the most vulnerable aspect of the target network. The proof is as below:

Proof: Assume $u_1(x_1)$ is the minimum value of the set $\{u_i(x_i)\}$ and ω_1 is the weight assigned to $u_1(x_1)$. If ω_1 become bigger, the value of:

$$U(\mathrm{X})=\prod_{i=1}^{n}[u_i(x)]^{\omega_i}$$

will be smaller.

Assume ω_1 has increased $\Delta\omega$ ($\Delta\omega>0$) and becomes:

$$\omega_1+\Delta\omega$$

The value of remaining ω_i must decrease. Assume they become:

$$\omega_i-\Delta\omega_i(\Delta\omega_i>0)$$

respectively. Note that:

$$\Delta\omega=\sum_{i=2}^{n}\Delta\omega_i$$

The new value of U(**x**) is:

$$U(\mathrm{X})=u_1(x_1)^{\omega_1+\Delta\omega}\prod_{i=2}^{n}[u_i(x_i)]^{\omega_i-\Delta\omega_i}$$

$$=\frac{u_1(x_1)^{\Delta\omega}}{\prod_{i=2}^{n}u_i(x_i)^{\Delta\omega_i}}\prod_{i=1}^{n}[u_i(x_i)]^{\omega_i}$$

$$<\frac{u_1(x_1)^{\Delta\omega}}{\prod_{i=2}^{n}u_1(x_1)^{\Delta\omega_i}}\prod_{i=1}^{n}[u_i(x_i)]^{\omega_i} \tag{6}$$

$$=\frac{u_1(x_1)^{\Delta\omega}}{u_1(x_1)^{\sum_{i=2}^{n}\Delta\omega_i}}\prod_{i=1}^{n}[u_i(x_i)]^{\omega_i}$$

$$= \prod_{i=1}^{n} \left[u_i \left(x_i \right) \right]^{\omega_i}$$

We can clearly see that with bigger value of ω_1, the value of aggregation function is smaller than the previous value. In the real scenario, if a user assigns big weight for one network utility value and unfortunately the corresponding utility is very small, this network should not be selected logically. After the proof, we can declare that the multi-criteria utility function above is suitable under such circumstance.

B. LBUF Algorithm Parameters and the Proof of Its Convergence

In the previous sections we successfully solve the problem of which network should be selected: just choose one with the maximum value of the multi-criteria utility function. In this section we will discuss the load balance based on the utility which is already calculated. Firstly we define several variables which will be used later.

1) Definition of parameters: Denote that l_i is the current amount of loads of network i and $l_i^{\max}$ is the maximum amount of loads of network i. Then the load rate is defined as:

$$\rho_i = \frac{l_i}{l_i^{\max}} \tag{7}$$

For the sake of real scenario such as LTE, l_i can be replaced by the resource blocks that are being used now and $l_i^{\max}$ can be replaced by the maximum available resource blocks.

Denote that parameter S is the variance of the set $\{\rho_i\}$, the expression of S is as below:

$$S = \frac{\sum_{i=1}^{n} \left(\rho_i - \mathrm{E}\rho \right)^2}{n} \tag{8}$$

where

$$\mathrm{E}\rho = \frac{\sum_{i=1}^{n} \rho_i}{n} \tag{9}$$

S_t is the threshold value of S. Once S is bigger than S_t, which implies that the load of the whole network is unbalanced, the load balance strategy should be triggered. [4] just provided a fixed value of S_t, which cannot meet the demand of the future network. We propose a self-adaptive threshold, the specific expression is as below:

$$S_t = \exp\left(-\alpha \frac{\sum_{i=1}^{n} l_i}{\sum_{i=1}^{n} l_i^{\max}} \right) + C \tag{10}$$

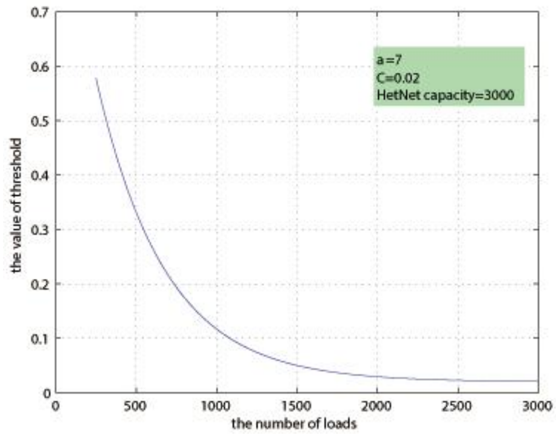

Fig. 3. Self-adaptive threshold.

α is a parameter that describes how balanced we want the loads to be. When the load in the HetNet is expected to be more balanced, a bigger value of α is required, and vise versa.

C is a constant value to prevent S_t from decreasing to zero. When there are too many users in the whole network, C represents the max load variance that the HetNet can endure.

$\sum_{i=1}^{n} l_i^{\max}$ can be explained as the capacity of the whole HetNet.

The image of the self-adaptive threshold S_t is shown in Fig. 3.

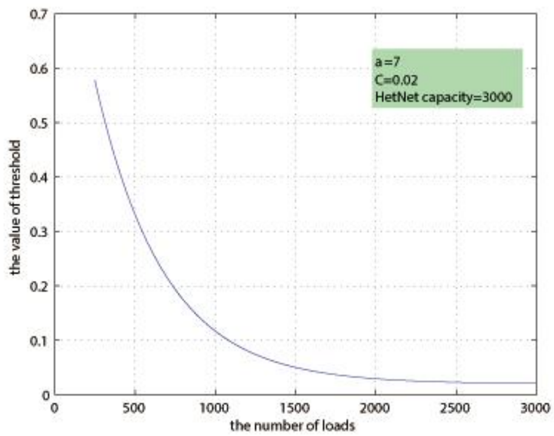

Fig. 3. Self-adaptive threshold.

When the load is slight($\rho_i < 0.7$), we don't control the variance because slight loads will not cause serious unbalance. At this point users are allowed accessing the network with the maximum utility value to maximize their profit. When some network is going to be overload, the threshold tends to limit the accessing while it is still relatively bigger. When the load is heavy, threshold would become small and the load balance mechanism would be frequently triggered.

2) The proof of the convergence of LBUF algorithm: As stated above, after users calculate each network's utility, they attempt to access the one with maximum utility value. The instantaneous variance S will be calculated. If $S < S_t$, the user is allowed accessing the target network. If $S \geq S_t$, the access requirement will be rejected by the controller and [4] provided an efficient method based on the margin. In this paper, the margin represents:

$$U(\mathrm{X}) - \rho_i \tag{11}$$

When $S > S_t$, the controller will make the user access one of the remaining networks with the maximum value of (14).

Proof: Denote:

$$S_0 = \frac{\sum_{i=1}^{n} (\rho_i - E_0)^2}{n} \tag{12}$$

as the variance before the access of one user who is to disturb the balance between the networks, where

$$E_0 = \frac{\sum_{i=1}^{n} \rho_i}{n} \tag{13}$$

Parameter S_1 is the variance after the access of that user. Its formulation is as below:

$$S_1 = \frac{\sum_{i=1, i \neq j}^{n} (\rho_i - E_1)^2 + (\rho_{j'} - E_1)^2}{n} \tag{14}$$

where j is the number of the suboptimum network; ρ_j is the load rate of network j before this time of access; $\rho_j^{'}$ is the load rate after that access hence the its expression is:

$$\rho_{j'} = \rho_j + \frac{1}{l_j^{\max}} \tag{15}$$

E_1 is the mean value after this access therefore:

$$E_1 = \frac{\sum_{i=1, i \neq j} n\rho_i + \rho_{j'}}{n} \tag{16}$$

We will prove that when $E_1 > \rho_j$, $S_0 > S_1$.

Proof:

$$S_1 - S_0 = \frac{\sum_{i=1}^{n} (\rho_i - E_0)^2}{n}$$

$$- \frac{\sum_{i=1, i \neq j}^{n} (\rho_i - E_1)^2 + (\rho_j^{'} - E_1)^2}{n}$$

$$= \frac{1}{n} (\sum_{i=1}^{n} \rho_i^2 + nE_0^2 - 2E_0 \sum_{i=1}^{n} \rho_i$$

$$- (\sum_{i=1}^{n} \rho_i^2 + \frac{1}{(l_j^{\max})^2} + 2\frac{\rho_j}{l_j^{\max}}$$

$$+ nE_1^2 - 2E_1 (\sum_{i=1}^{n} \rho_i + \frac{1}{l_j^{\max}})))$$

$$= -\frac{1}{(l_j^{\max})^2} - 2\frac{\rho_j}{l_j^{\max}} + n(E_0^2 - E_1^2)$$

$$- 2E_0 \sum_{i=1}^{n} \rho_i + 2E_1 \sum_{i=1}^{n} \rho_i + \frac{2E_1}{l_j^{\max}}$$

$$= -\frac{1}{(l_j^{\max})^2} - 2\frac{\rho_j}{l_j^{\max}} + \frac{\left(\sum_{i=1}^{n} \rho_i\right)^2}{n}$$

$$- \frac{\left(\sum_{i=1}^{n} \rho_i + \frac{1}{l_j^{\max}}\right)^2}{n}$$

$$+ 2\sum_{i=1}^{n} \rho_i (E_1 - E_0) + \frac{2E_1}{l_j^{\max}}$$

$$= -\frac{1}{(l_j^{\max})^2} - \frac{1}{n(l_j^{\max})^2} + \frac{2E_1}{l_j^{\max}} - \frac{2\rho_j}{l_j^{\max}} \tag{17}$$

Here exists an approximate calculation: considering $l_j^{\max}$ is usually as big as several hundreds even thousands, therefore:

$$\frac{1}{(l_j^{\max})^2} << \frac{1}{l_j^{\max}}$$

We neglect $-\frac{1}{(l_j^{\max})^2} - \frac{1}{n(l_j^{\max})^2}$ in (20), so we mainly need to calculate:

$$\frac{2E_1}{l_j^{\max}} - \frac{2\rho_j}{l_j^{\max}}$$

i.e.,

$$E_1 - \rho_j$$

If we want $S_0 > S_1$, we require $E_1 > \rho_j$. This condition is easy to obtain because the mean value E_1 will certainly increase with the access of users and due to the suboptimum network has the maximum value of $U(x) - \rho_j$, the ρ_j is relatively small compared with other network's load rate. So this load balance method is feasible and we will see its satisfying performance in the simulation section.

3) procedure of LBUF algorithm: The LBUF algorithm is divided into 3 parts: users generator, utility calculation and load balance. The pseudo-code is shown below:

Algorithm 1 Users Generator

```
for i ← 1 to lenght(load) do
    r ← random()
    a ← random()
    while a < 0.7 do
        a ← random()
    end while
    b ← (1−a)/3
    if r < 0.10 then
        load(i).w_data_rate ← a
        the other load(i).w ← b
    else
        if 0.10 ≤ r < 0.35 then
            load(i).w_cost ← a
            the other load(i).w ← b
        else
            if 0.35 ≤ r < 0.75 then
                load(i).w_power_consumption ← a
                the other load(i).w ← b
            else
                load(i).w_link_quality ← a
                the other load(i).w ← b
            end if
        end if
    end if
end for
```

Algorithm 2 Utility Calculation

for $i \leftarrow 1$ to $length(load)$ do

$$load(i).utility\_LTE \leftarrow \prod_{i=1}^{4}\left[u_i\left(x_i^{LTE}\right)\right]^{\omega_i}$$

$$load(i).utility\_WiFi \leftarrow \prod_{i=1}^{4}\left[u_i\left(x_i^{WiFi}\right)\right]^{\omega_i}$$

$$load(i).utility\_WiMAX \leftarrow \prod_{i=1}^{4}\left[u_i\left(x_i^{WiMAX}\right)\right]^{\omega_i}$$

end for

Algorithm 3 Load Balance

for $i \leftarrow 1$ to $length(load)$ do

find the optimized network α with maximum utility value among three networks

$l_\alpha \leftarrow optimized\_network.current\_load$

$l_\alpha \leftarrow l_\alpha + 1$

for $i \leftarrow 1$ to 3 do

$\rho_i \leftarrow network\_i.current\_load / network\_i.\max\_load$

end for

$$E \leftarrow \frac{1}{n}\sum_{i=1}^{3}\rho_i$$

$$S \leftarrow \frac{1}{n}\sum_{i=1}^{3}(\rho_i - E)$$

$$S_t \leftarrow \exp\left(-\alpha\frac{\sum_{i=1}^{3} network\_i.current\_load}{\sum_{i=1}^{3} network\_i.\max\_load}\right) + C$$

if $(S > S_t \ \& \ \rho_i > 0.7)$ then

find the suboptimum network with maximum $U(X) - \rho_i$ among the remaing networks

$l_\alpha \leftarrow l_\alpha - 1$

$l_\beta \leftarrow suboptimum\_network.current\_load$

$l_\beta \leftarrow l_\beta + 1$

end if

end for

IV. Simulation Parameters and Results

In this section, we will show the perfect performance of LBUF algorithm. The simulation platform is MatLab.

TABLE I: NETWORK PARAMETERS

	LTE	WiFi	WiMAX
Cost(cents/MB)	110	10	45
Data rate(Mbps)	100	70	75
Power Consumption	2.5	4	3.5
Link quality	7	8	5.5
Max loads	750	600	450

The HetNet scenario consists of LTE, WiFi and WiMax. We mainly consider four factors that will influence the network selection: money cost, data rate, power consumption, and link quality. The detailed parameters of them is listed in Table I. The value of link quality and power consumption are relative value without unit.

The specific parameters used in utility function is shown in Table II (γ is always equal to $\frac{2\left(x_\beta - x_m\right)}{x_m - x_\alpha}$).

The parameters used in the self-adaptive threshold of variance are: $C = 0.003$ $\alpha = 35$.

We classify users (i.e. loads) into four groups. Their preference weight is generated by Algorithm 1.

We divide the whole simulation process into 400 slots. In each slot, there is a random number (between 0 and 12, discrete uniform distribution) of users who attempt to access the network. After the 200th slot, there is random number (between 0 and 12, discrete uniform distribution) of users who leave the network. Under such scenario, the results of the simulation for comparison are as following.

Fig. 4 and Fig. 5 show the variance of the load rate in the situation of the self-adaptive threshold and the fixed one. They both fluctuated remarkably and converge to the corresponding threshold. We can clearly see that in early, the variance is relatively bigger if the threshold is self-adaptive, which is because at that time we didn't limit the users' accessing.

Fig. 6 and Fig. 7 show the load rate of the three types of network in the system. In Fig. 6, it is obviously that the load rate comes into a platform when it reaches 0.7, because at that time the threshold comes into operation and the users accessed to the suboptimal network. Comparing with Fig. 7, apparently tol.

he self-adaptive threshold has better performance of contr

TABLE II: PARAMETERS USED IN UTILITY FUNCTIONS

	$\left[x_\alpha, x_\beta\right]$	x_m	ζ
U_{cost}	$[5,150]$	100	5
$U_{data\ \ rate}$	$[0,100]$	50	4
$U_{power\ \ consumption}$	$[1,6]$	3.5	3.9
$U_{link\ \ quality}$	$[1,10]$	5	4

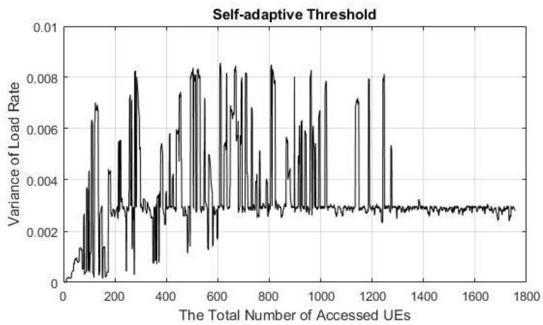

Fig. 4. Variance of load rate with self-adaptive threshold.

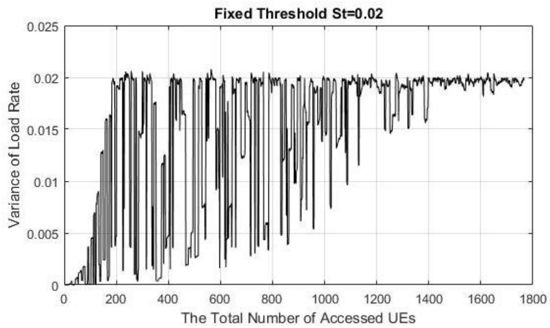

Fig. 5. Variance of load rate with fixed threshold.

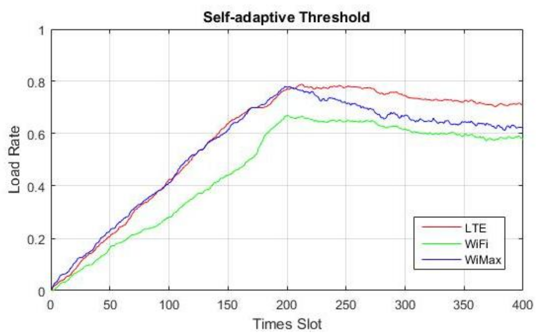

Fig. 6. Variance of load rate with self-adaptive threshold.

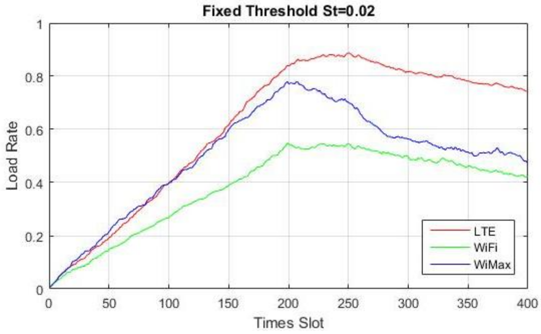

Fig. 7. Variance of load rate with fixed threshold.

We can find the same conclusion in Fig.8 and Fig.9. They show the max value of the utility of the network which users access at each time slot. In comparison with the fixed threshold algorithm, the self-adaptive threshold algorithm hits local minimums at lower frequency and higher user utilities.

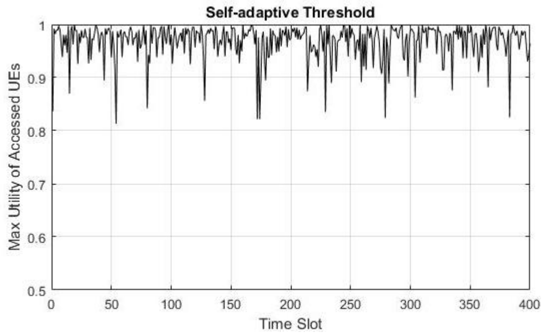

Fig. 8. Variance of load rate with self-adaptive threshold.

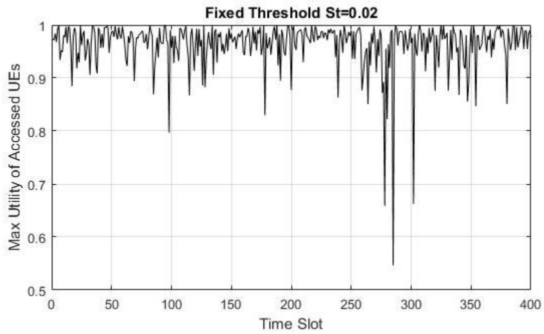

Fig. 9. Variance of load rate with fixed threshold.

V. CONCLUSIONS

We have proposed an algorithm called LBUF to balance the load in HetNet. Not only the load level of the network is considered, the preference of user is also highlighted. In LBUF algorithm, we suggested a self-adaptive threshold, which will automatically change according to the current number of loads in HetNet. Obviously LBUF algorithm needs global data of HetNet which traditional network cannot provide. Therefore, our method is suitable for SDN, where the LBUF algorithm can be written in the SDN controller. In the future, we plan to do further simulations to get critical parameters which are essential in the real deployment such as Radio

Link Failure (RLF) rate, Handover (HO) failure rate and Ping-pong rate.

REFERENCES

[1] Y. Yang, P. Li, X. Chen, and W. Wang, "A high-efficient algorithm of mobile load balancing in lte system," in *Proc. Vehicular Technology Conference (VTC Fall)*, Sept. 2012, pp. 1-5.

[2] C. Yang, H. Li, and Y. Dai, "Net utility-oriented mobility load balancing in self-optimizing networks," in *Proc. IEEE International Conference on Computer and Information Technology*, Sept. 2014, pp. 14-19.

[3] J. Sheng, Z. Yang, L. R. Tang, and J. H. Hao, "A novel load balancing algorithm based on utility functions and fuzzy logic in heterogeneous wireless networks," in *Proc. 9th International Conference on Fuzzy Systems and Knowledge Discovery*, May 2012, pp. 414-418.

[4] W. Tang and Q. Liao, "An sdn-based approach for load balance in heterogeneous radio access networks," in *IEEE Symposium on Computer Applications and Communications*, July 2014, pp. 105-108.

[5] H. Dong, X. Xia, L. Jia, Y. Tian, H. Li, J. Tang, and Y. Qin, "Quantitatively analysis of train communication network based on multi-attribute utility function," in *IEEE 17th International Conference on Intelligent Transportation Systems*, Oct. 2014, pp. 2330-2335.

[6] Q. T. Nguyen-Vuong, Y. Ghamri-Doudane, and N. Agoulmine, "On utility models for access network selection in wireless heterogeneous networks," in *Proc. Network Operations and Management Symposium, IEEE*, April 2008, pp. 144-151.

[7] Q. T. Nguyen-Vuong, N. Agoulmine, E. H. Cherkaoui, and L. Toni, " Multicriteria optimization of access selection to improve the quality of experience in heterogeneous wireless access networks," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 4, pp. 1785-1800, May 2013.

[8] D. Kreutz, F. M. V. Ramos, P. E. Verlssimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," *Processing of the IEEE*, vol. 103, no. 1, pp. 14-76, Jan. 2015.

Qing Wang received the B.Eng. and M.Eng. degrees in electronic engineering and the Ph.D. degree in communication engineering from Tianjin University, Tianjin, China, in 2004, 2007 and 2010, respectively, and where she is currently an associate professor. From October 2007 to April 2009, she was a Visiting Student with the School of Electrical and Electronic Engineering, Nanyang Techno-logical University, Singapore. Her current research interests include wireless communication, passive radar, radio propagation and channel modeling.

Minjiong Zhu received the B.Eng. Degrees in electronic information engineering from Tianjin University, Tianjin, China, in 2014. Now he is studying for his master degree in Tianjin University. His current research interests include software-defined network and wireless communication.

Yongrun Yu received the bachelor degree from Tianjin University, China, in 2016. Now he is studying for his master degree in Southeast University, China. His research interest is mobile communication network.

Maode Ma received his Ph.D.degree in Department of Computer Science from Hong Kong University of Science and Technology in 1999. Now, Dr.Ma is an Associate Professor in the School of Electrical and Electronic Engineering at Nanyang Technological University in Singapore. He has extensive research interests including network security and wireless networking. Dr.Ma has over 300 international academic publications including more than 130 journal papers and over 160 conference papers. He currently serves as the Editor-in-Chief of International Journal of Computer and Communication Engineering and International Journal of Electronic Transport. He also serves as a Senior Editor for IEEE Communications Surveys and Tutorials, and an Associate Editor for International Journal of Network and Computer Applications, International Journal of Security and Communication Networks, International Journal of Wireless Communication and Mobile Computing and International Journal of Communication Systems. Dr.Ma is the Fellow of IET and a senior member of IEEE Communication Society and IEEE Education Society. He is the Chair of the IEEE Education Society, Singapore Chapter. He is serving as an IEEE Communication Society Distinguished Lecturer from 2013 to 2016.

Cui Xin received the bachelor degree from Tianjin University, China, in 2016. Now she is studying for her master degree in Tianjin University. Her current research interest is wireless communication.