A Robust Optimal Rate Allocation Algorithm and Pricing Policy for Hybrid Traffic in 4G-LTE

Ahmed Abdel-Hadi and Charles Clancy
ECE, Virginia Tech, Arlington, VA, 22203, USA
{aabdelhadi, tcc}@vt.edu

Abstract—**In this paper, we consider resource allocation optimization problem in the fourth generation long-term evolution (4G-LTE) with elastic and inelastic real-time traffic. Mobile users are running either delay-tolerant or real-time applications. The users applications are approximated by logarithmic or sigmoidal-like utility functions. Our objective is to allocate resources according to the utility proportional fairness policy. Prior utility proportional fairness resource allocation algorithms fail to converge for high-traffic situations. We present a robust algorithm that solves the drawbacks in prior algorithms for the utility proportional fairness policy. Our robust optimal algorithm allocates the optimal rates for both high-traffic and low-traffic situations. It prevents fluctuation in the resource allocation process. In addition, we show that our algorithm provides traffic-dependent pricing for network providers. This pricing could be used to flatten the network traffic and decrease the cost per bandwidth for the users. Finally, numerical results are presented on the performance of the proposed algorithm.**

I. Introduction

In recent years, there has been a significant growth in the demand for higher data rates to support the transition from voice-only communications to multimedia communications in mobile systems, e.g. 4G-LTE. This demand motivates numerous research efforts to optimally allocate the available limited bandwidth resources for users seeking better quality-of-service (QoS). One aspect of improving resource allocation and achieving better QoS is to use network utility optimization. Network utility optimization was initially used for rate allocation in wired networks such as the Internet [1], [2]. The utility functions used in the Internet are delay-tolerant utility functions that can be mathematically represented by logarithmic functions. In recent research work, e.g. [3], [4], network utility optimization is used to allocate resources in wireless networks for real-time applications. Real-time applications such as voice-over-IP (VoIP) and video streaming are mathematically represented by sigmoidal-like utility functions with different parameters for different real-time applications [5]. The majority of prior work done on wireless network utility optimization for real-time applications only provides approximations of the optimal rate allocation.

In [6], the authors used resource allocation optimization problem with utility proportional fairness objective function (i.e the objective is to provide fair utility percentage for all the users). The authors included hybrid traffic in their model where users utility functions are either logarithmic or sigmoidal-like functions (i.e. correspond to delay-tolerant or real-time applications). Despite the absence of concavity in the sigmoidal-like utility functions, the authors have reformulated the optimization problem and proven that it is a convex optimization problem. Therefore, a tractable global optimal solution exists. Using duality, the authors presented a distributed iterative algorithm for allocating the optimal rates to users. This rate allocation is fair with respect to utility percentage resulting in allocation priority given to users with real-time applications over users with delay-tolerant applications. Meanwhile, the algorithm ensures that no user receives zero rate (i.e. no user is dropped). Therefore, the algorithm ensures minimum QoS for all network subscribers and priority to real-time application users who are paying more for better QoS.

The distributed rate allocation algorithm presented in [6] converges to the optimal rate only when the maximum available rate by the eNodeB exceeds the sum of rates needed to achieve 50 % utility percentage for all the real-time application users. Therefore, the algorithm doesn't converge for an eNodeB with scarce bandwidth resources with respect to the number of users and their utilities. In this paper, we analyze this situation which occurs frequently during peak network usage periods of the day. Our algorithm presents a more robust algorithm that converges for both scarce and abundant bandwidth resources. In addition to allocating the optimal rates, we show that our algorithm provides a traffic-dependent pricing approach that could be used by network providers.

A. Related Work

In [7], [8], the authors presented a non-convex optimization problem for maximization of utility functions in wireless networks. They used both hybrid utility functions and presented the algorithm to solve it optimally when the duality gap is zero. They included a fair allocation heuristic algorithm to ensure network stability which resulted in a high aggregated utility.

In [9], the authors proposed a utility max-min fairness resource allocation for users with elastic and inelastic traffic. In [10], the authors proposed a utility proportional fair optimization problem for high-SINR wireless networks using utility max-min architecture. They compared their algorithm to the traditional bandwidth proportional fair algorithms and presented a closed form solution that prevents fluctuation.

In [3], the authors presented a distributed power allocation algorithm for cellular systems. They used non-concave sigmoidal-like utility functions. The proposed algorithm approximates the global optimal solution and can drop users to

978-1-4577-1348-4/13/$31.00 ©2013 IEEE

maximize the overall system utilities, therefore, it does not guarantee minimum QoS for all users.

B. Our Contributions

Our contributions in this paper are summarized as:

- We consider the utility proportional fairness resource allocation optimization problem for both elastic and inelastic traffic. We analyze the convergence of the distributed rate allocation algorithm that is presented in [6]. We show that it doesn't converge to the optimal rates in high-traffic periods (i.e. resources are scarce with respect to number of active users).
- We present a robust distributed rate allocation algorithm that converges to the optimal rates for high-traffic and low-traffic periods.
- We present a pricing policy for network providers that can flatten traffic load on the network and decrease the overall service cost to subscribers.

The remainder of this paper is organized as follows. Section II presents the problem formulation. Section III analyzes the rate allocation algorithm in [6] and discusses its convergence. In Section IV, we present our distributed rate allocation algorithm. Section V discusses simulation setup and provides quantitative results along with discussion. Section VI concludes the paper.

II. Problem Formulation

We consider a system model that is similar to [6] where we have a single cell 4G-LTE mobile system consisting of a single evolved Node B (eNodeB) and M user equipments (UE)s. The rate allocated by the eNodeB to i^{th} UE is given by r_i. Each UE has its own utility function $U_i(r_i)$ that corresponds to the type of traffic being handled by it. Our objective is to determine the optimal rates the eNodeB allocates to the UEs. We assume the utility function $U_i(r_i)$ of the i^{th} UE to be a strictly concave or a sigmoidal-like function. The utility functions $U(r)$ have the following properties:

- $U(0) = 0$ and $U(r)$ is an increasing function of r.
- $U(r)$ is twice continuously differentiable in r and bounded above.

In our model, we use the normalized sigmoidal-like utility function, as in [3], to represent real-time applications running on the UEs that can be expressed as

$$U(r) = c\Big(\frac{1}{1+e^{-a(r-b)}} - d\Big) \tag{1}$$

where $c = \frac{1+e^{ab}}{e^{ab}}$ and $d = \frac{1}{1+e^{ab}}$. So, it satisfies $U(0) = 0$ and $U(\infty) = 1$. The inflection point of the normalized sigmoidal-like function is at $r^{\text{inf}} = b$. In addition, we use the normalized logarithmic utility function, as in [10], to represent delay-tolerant applications running on the UEs that can be expressed as

$$U(r) = \frac{\log(1+kr)}{\log(1+kr_{max})} \tag{2}$$

where r_{max} is the maximum required rate for the user to achieve 100% utility percentage and k is the rate of increase of utility percentage with the allocated rate. So, it satisfies $U(0) = 0$ and $U(r_{max}) = 1$. The inflection point of normalized logarithmic function is at $r^{\text{inf}} = 0$.

The basic formulation of the resource allocation problem is given by the following optimization problem:

$$\begin{aligned} \max_{\mathbf{r}} \quad & \prod_{i=1}^{M} U_i(r_i) \\ \text{subject to} \quad & \sum_{i=1}^{M} r_i \leq R \\ & r_i \geq 0, \quad i = 1,2,...,M. \end{aligned} \tag{3}$$

where R is the maximum achievable rate of the eNodeB and $\mathbf{r} = \{r_1, r_2, ..., r_M\}$ are the rates allocated to the UEs.

The existence of a tractable global optimal solution for the optimization problem (3) is proven in [6].

III. Rate Allocation Algorithm and its Fluctuation

A. Rate Allocation Algorithm

In [6], the authors proposed a rate allocation algorithm to allocate the optimal rates for the optimization problem in equation (3). The algorithm is an iterative distributed algorithm, which is a modified version of Frank Kelly algorithm in [1].

Algorithm 1 UE Algorithm in [6]

```
Send initial bid w_i(1) to eNodeB
loop
  Receive shadow price p(n) from eNodeB
  if STOP from eNodeB then
    Calculate allocated rate r_i^opt = w_i(n) / p(n)
  else
    Solve r_i(n) = arg max_{r_i} ( log U_i(r_i) - p(n) r_i )
    Send new bid w_i(n) = p(n) r_i(n) to eNodeB
  end if
end loop
```

The algorithm is divided into two parts, Algorithm (1) which runs on the UE side and Algorithm (2) which runs on the eNodeB side. The i^{th} UE solves for its bid $w_i(n)$ and sends it to the eNodeB. The eNodeB calculates the shadow price $p(n)$ and sends it to all UEs. Each UE uses the shadow price to recalculate its new bid until $|w_i(n) - w_i(n-1)|$ is less than a pre-specified threshold δ.

Now, we show the fluctuation in Algorithm (1) and (2) (i.e. convergence analysis) when running on an eNodeB with scarce bandwidth resources with respect to the number of users and the shape of their utility functions.

B. Convergence Analysis

In this section, we present the convergence analysis of Algorithm (1) and (2) for different values of R.

Lemma III.1. *For sigmoidal-like utility function $U_i(r_i)$, the slope curvature function $\frac{\partial \log U_i(r_i)}{\partial r_i}$ has an inflection point at $r_i = r_i^s \approx b_i$ and is convex for $r_i > r_i^s$.*

Algorithm 2 eNodeB Algorithm in [6]

loop
  Receive bids $w_i(n)$ from UEs {Let $w_i(0) = 0 \ \forall i$}
  if $|w_i(n) - w_i(n-1)| < \delta \ \forall i$ **then**
    STOP and allocate rates (i.e r_i^{opt} to user i)
  else
    Calculate $p(n) = \frac{\sum_{i=1}^{M} w_i(n)}{R}$
    Send new shadow price $p(n)$ to all UEs
  end if
end loop

Proof: For the sigmoidal-like function $U_i(r_i) = c_i\Big(\frac{1}{1+e^{-a_i(r_i-b_i)}} - d_i\Big)$, let $S_i(r_i) = \frac{\partial \log U_i(r_i)}{\partial r_i}$ be the slope curvature function. Then, we have that

$$\frac{\partial S_i}{\partial r_i} = \frac{-a_i^2 d_i e^{-a_i(r_i-b_i)}}{c_i\Big(1-d_i(1+e^{-a_i(r_i-b_i)})\Big)^2} - \frac{a_i^2 e^{-a_i(r_i-b_i)}}{\Big(1+e^{-a_i(r_i-b_i)}\Big)^2}$$

and

$$\frac{\partial^2 S_i}{\partial r_i^2} = \frac{a_i^3 d_i e^{-a_i(r_i-b_i)}(1-d_i(1-e^{-a_i(r_i-b_i)}))}{c_i\Big(1-d_i(1+e^{-a_i(r_i-b_i)})\Big)^3} + \frac{a_i^3 e^{-a_i(r_i-b_i)}(1-e^{-a_i(r_i-b_i)})}{\Big(1+e^{-a_i(r_i-b_i)}\Big)^3}. \tag{4}$$

We analyze the curvature of the slope of the natural logarithm of sigmoidal-like utility function. For the first derivative, we have $\frac{\partial S_i}{\partial r_i} < 0 \ \forall r_i$. The first term S_i^1 of $\frac{\partial^2 S_i}{\partial r_i^2}$ in equation (4) can be written as

$$S_i^1 = \frac{a_i^3 e^{a_i b_i}(e^{a_i b_i} + e^{-a_i(r_i-b_i)})}{(e^{a_i b_i} - e^{-a_i(r_i-b_i)})^3} \tag{5}$$

and we have

$$\lim_{r_i \to 0} S_i^1 = \infty, \text{ and } \lim_{r_i \to b_i} S_i^1 = 0 \text{ for } b_i \gg \frac{1}{a_i}. \tag{6}$$

For second term S_i^2 of $\frac{\partial^2 S_i}{\partial r_i^2}$ in equation (4), we have the following properties

$$S_i^2(b_i) = 0, \ S_i^2(r_i > b_i) > 0, \text{ and } S_i^2(r_i < b_i) < 0. \tag{7}$$

From equation (6) and (7), S_i has an inflection point at $r_i = r_i^s \approx b_i$. In addition, we have the curvature of S_i changes from a convex function close to origin to a concave function before the inflection point $r_i = r_i^s$ then to a convex function after the inflection point. ■

Corollary III.2. *If $\sum_{i=1}^{M} r_i^{inf} \ll R$ then Algorithm in (1) and (2) converges to the global optimal rates which correspond to the steady state shadow price $p_{ss} < \frac{a_{i_{\max}} d_{i_{\max}}}{1-d_{i_{\max}}} + \frac{a_{i_{\max}}}{2}$ where $i_{\max} = \arg\max_i b_i$.*

Proof: For the sigmoidal-like function $U_i(r_i) = c_i\Big(\frac{1}{1+e^{-a_i(r_i-b_i)}} - d_i\Big)$, the optimal solution is achieved by solving the optimization problem (3). In Algorithm (1), an important step to reach to the optimal solution is to solve the optimization problem $r_i(n) = \arg\max_{r_i}\Big(\log U_i(r_i) - p(n) r_i\Big)$ for every UE. The solution of this problem can be written, using Lagrange multipliers method, in the form

$$\frac{\partial \log U_i(r_i)}{\partial r_i} - p = S_i(r_i) - p = 0. \tag{8}$$

From equation (6) and (7) in Lemma III.1, we have the curvature of $S_i(r_i)$ is convex for $r_i > r_i^s \approx b_i$. The Algorithm in (1) and (2) is guaranteed to converges to the global optimal solution when the slope $S_i(r_i)$ of all the utility functions natural logarithm $\log U_i(r_i)$ are in the convex region of the functions, similar to analysis of logarithmic functions in [1] and [2]. Therefore, the natural logarithm of sigmoidal-like functions $\log U_i(r_i)$ converge to the global optimal solution for $r_i > r_i^s \approx b_i$. The inflection point of sigmoidal-like function $U_i(r_i)$ is at $r_i^{\text{inf}} = b_i$. For $\sum_{i=1}^{M} r_i^{\text{inf}} \ll R$, Algorithm in (1) and (2) allocates rates $r_i > b_i$ for all users. Since $S_i(r_i)$ is convex for $r_i > r_i^s \approx b_i$ then the optimal solution can be achieved by Algorithm (1) and (2). We have from equation (8) and as $S_i(r_i)$ is convex for $r_i > r_i^s \approx b_i$, that $p_{ss} < S_i(r_i = \max b_i)$ where $S_i(r_i = \max b_i) = \frac{a_{i_{\max}} d_{i_{\max}}}{1-d_{i_{\max}}} + \frac{a_{i_{\max}}}{2}$ and $i_{\max} = \arg\max_i b_i$. ■

Corollary III.3. *For $\sum_{i=1}^{M} r_i^{inf} > R$ and the global optimal shadow price $p_{ss} \approx \frac{a_i d_i e^{\frac{a_i b_i}{2}}}{1-d_i(1+e^{\frac{a_i b_i}{2}})} + \frac{a_i e^{\frac{a_i b_i}{2}}}{(1+e^{\frac{a_i b_i}{2}})}$, then the solution by Algorithm in (1) and (2) fluctuates about the global optimal solution.*

Proof: It follows from lemma III.1 that for $\sum_{i=1}^{M} r_i^{\text{inf}} > R \ \exists \ i$ such that the optimal rates $r_i^{\text{opt}} < b_i$. Therefore, if $p_{ss} \approx \frac{a_i d_i e^{\frac{a_i b_i}{2}}}{1-d_i(1+e^{\frac{a_i b_i}{2}})} + \frac{a_i e^{\frac{a_i b_i}{2}}}{(1+e^{\frac{a_i b_i}{2}})}$ is the optimal shadow price for optimization problem (3). Then, a small change in the shadow price $p(n)$ in the n^{th} iteration can lead the rate $r_i(n)$ (root of $S_i(r_i) - p(n) = 0$) to fluctuate between the concave and convex curvature of the slope curve $S_i(r_i)$ for the i^{th} user. Therefore, it causes fluctuation in the bid $w_i(n)$ sent to the eNodeB and fluctuation in the shadow price $p(n)$ set by eNodeB. Therefore, the iterative solution of Algorithm in (1) and (2) fluctuates about the global optimal rates r_i^{opt}. ■

Theorem III.4. *Algorithm in (1) and (2) does not converge to the global optimal solution for all values of R.*

Proof: It follows from Corollary III.2 and III.3 that Algorithm in (1) and (2) does not converge to the global optimal solution for all values of R. ■

C. Fluctuation Example

We consider an example of four users where two users run applications with sigmoidal-like utility functions and the other two users run applications with logarithmic utility functions. The sigmoidal-like utility functions parameters are $a = \{5, 0.5\}$ and $b = \{10, 20\}$, respectively. The logarithmic utility functions parameters are $k = \{15, 0.1\}$ and $r_{max} = 100$. We assume that the eNodeB maximum rate is $R = 25$, therefore $\sum_{i=1}^{4} r_i^{\text{inf}} = 30 > R = 25$, therefore we can't guarantee convergence with Algorithm in (1) and (2), as stated by Corollary III.3. In Figure 1, we show that the shadow price $p(n)$ fluctuates between a concave and convex curvature

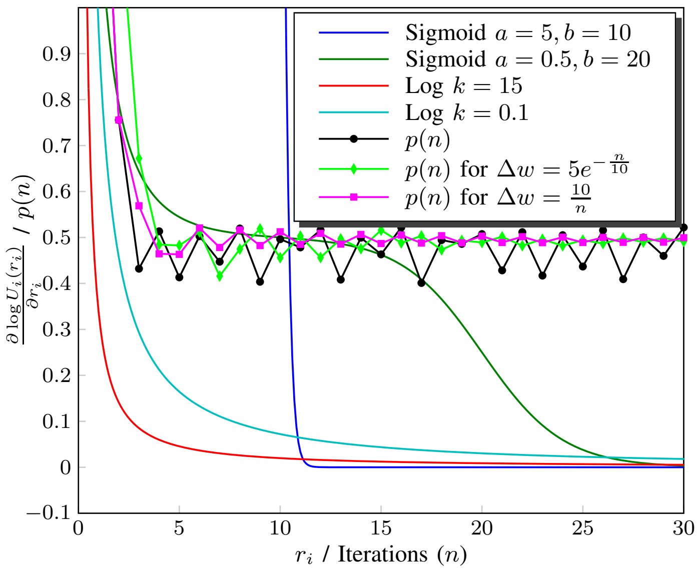

Fig. 1. The $\frac{\partial \log U_i(r_i)}{\partial r_i}$ curve of fluctuation example in Section III-C, the shadow price $p(n)$ from Algorithm in (1) and (2), and the shadow price $p(n)$ of Algorithm in (3) and (4) with $\Delta w = 5e^{-\frac{n}{10}}$ and $\Delta w = \frac{10}{n}$ for $R = 25$ (i.e. $\sum r_i^{\text{inf}} > R$).

of the $\frac{\partial \log U_i(r_i)}{\partial r_i}$ curve. The fluctuation in the shadow price $p(n)$ causes fluctuation in the allocated rates and hinders the convergence to the optimal rates. Therefore, the optimal rate allocation is not achievable by Algorithm in (1) and (2).

IV. Our Distributed Algorithm

In this section, we present our robust algorithm to ensure the rate allocation algorithm in [6] converges for all values of the eNodeB maximum rate R. Our algorithm allocate rates coincide with the Algorithm in (1) and (2) for $\sum r_i^{\text{inf}} > R$. For $\sum r_i^{\text{inf}} \ll R$, our algorithm avoids the fluctuation in the non-convergent region discussed in the previous section. This is achieved by adding a convergence measure $\Delta w(n)$ that senses the fluctuation in the bids w_i. In case of fluctuation, our algorithm decreases the step size between the current and the previous bid $w_i(n) - w_i(n-1)$ for every user i using *fluctuation decay function*. The fluctuation decay function could be in the following forms:

- *Exponential function*: It takes the form $\Delta w(n) = l_1 e^{-\frac{n}{l_2}}$.
- *Rational function*: It takes the form $\Delta w(n) = \frac{l_3}{n}$.

where l_1, l_2, l_3 can be adjusted to change the rate of decay of the bids w_i. The new algorithm with the fluctuation decay function is in Algorithm (3) and (4).

Remark IV.1. *The fluctuation decay function can be included in Algorithm (3) of the UE or Algorithm (4) of the eNodeB.*

In our model, we add the decay part in Algorithm (3) of the UE. In Figure 1, we show the new shadow price $p(n)$ of the fluctuation example in Section III-C when using Algorithm in (3) and (4). The shadow price $p(n)$ fluctuation decreases with every iteration n and converges to the optimal shadow price that corresponds to the optimal rates. A detailed example is given in the simulation section (Section V).

Algorithm 3 Our UE Algorithm

```
Send initial bid w_i(1) to eNodeB
loop
    Receive shadow price p(n) from eNodeB
    if STOP from eNodeB then
        Calculate allocated rate r_i^opt = w_i(n)/p(n)
    else
        Calculate new bid w_i(n) = p(n) r_i(n)
        if |w_i(n) − w_i(n−1)| > Δw(n) then
            w_i(n) = w_i(n−1) + sign(w_i(n) − w_i(n−1)) Δw(n)
            {Δw = l_1 e^(−n/l_2) or Δw = l_3/n}
        end if
        Send new bid w_i(n) to eNodeB
    end if
end loop
```

Algorithm 4 Our eNodeB Algorithm

```
loop
    Receive bids w_i(n) from UEs {Let w_i(0) = 0 ∀i}
    if |w_i(n) − w_i(n−1)| < δ ∀i then
        STOP and calculate rates r_i^opt = w_i(n)/p(n)
    else
        Calculate p(n) = (Σ_{i=1}^{M} w_i(n))/R
        Send new shadow price p(n) to all UEs
    end if
end loop
```

V. Simulation Results

Algorithm in (3) and (4) was applied to various logarithmic and sigmoidal-like utility functions with different parameters using MATLAB. Our simulation results showed convergence to the optimal global rates for all values of the eNodeB rate R. In this section, we use simulation setting and parameters similar to [6], we present the simulation results of six utility functions corresponding to six UEs shown in Figure 2. We use three normalized sigmoidal-like functions that are expressed by equation (1) with different parameters, $a = 5$, $b = 10$ which is an approximation to a step function at rate $r = 10$ (e.g. VoIP), $a = 3$, $b = 20$ which is an approximation of an adaptive real-time application with inflection point at rate

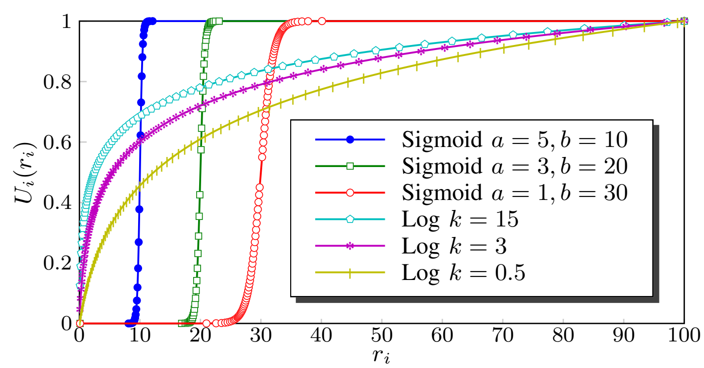

Fig. 2. The users utility functions $U_i(r_i)$ used in the simulation (three sigmoidal-like functions and three logarithmic functions).

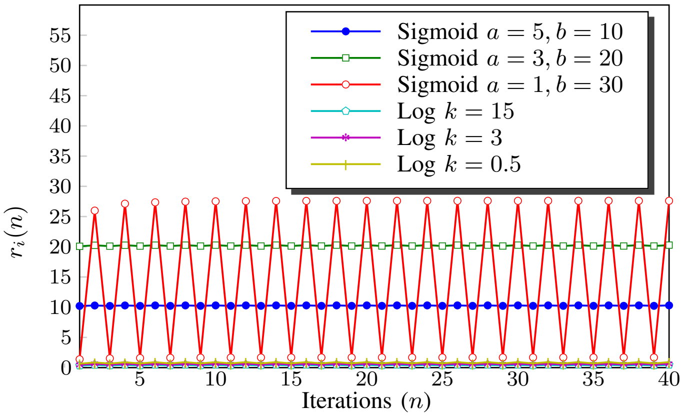

Fig. 3. The rates convergence $r_i(n)$ of Algorithm in (1) and (2) with number of iterations n for different users and $R = 45$.

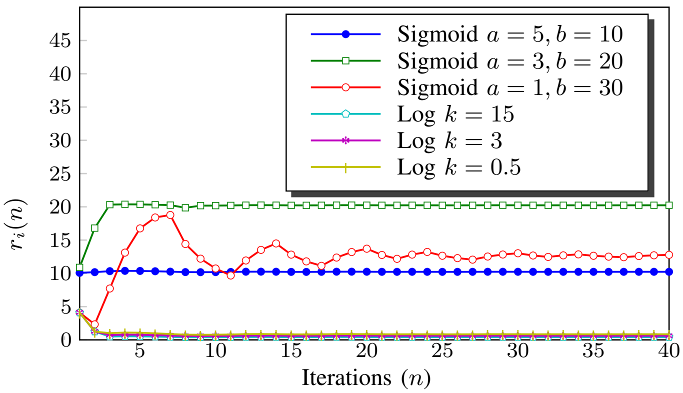

Fig. 4. The rates convergence $r_i(n)$ of Algorithm in (3) and (4) with number of iterations n for different users and $R = 45$.

$r = 20$ (e.g. standard definition video streaming), and $a = 1$, $b = 30$ also is an approximation of an adaptive real-time application with inflection point at rate $r = 30$ (e.g. high definition video streaming). We use three logarithmic functions that are expressed by equation (2) with $r_{max} = 100$ and different k_i parameters which are approximation for delay-tolerant applications (e.g. FTP). We use $k = \{15, 3, 0.5\}$.

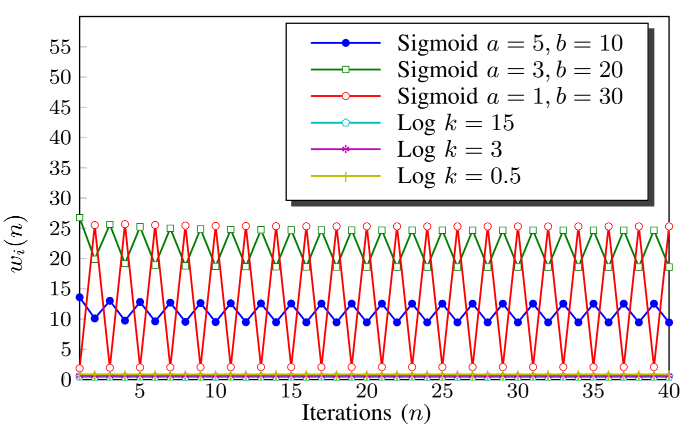

Fig. 5. The bids convergence $w_i(n)$ of Algorithm in (1) and (2) with number of iterations n for different users and $R = 45$.

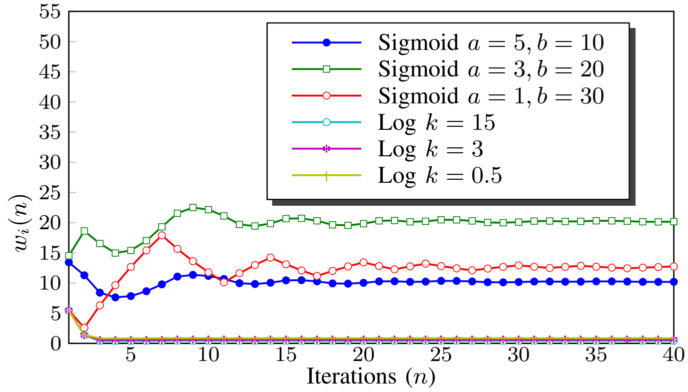

Fig. 6. The bids convergence $w_i(n)$ of Algorithm in (3) and (4) with number of iterations n for different users and $R = 45$.

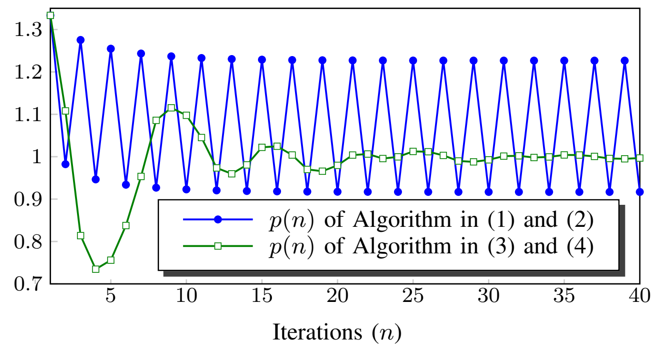

Fig. 7. The shadow price $p(n)$ convergence with the number of iterations n.

A. Convergence Dynamics for R =45

In the following simulations, we set $R = 45$ and number of iterations $n = 40$. Here, we choose the total eNodeB rate R to be less than the sum of real-time application users inflection points $\sum b_i$. Therefore, Algorithm in (1) and (2) does not converge in this region. In Figure 3, we show the

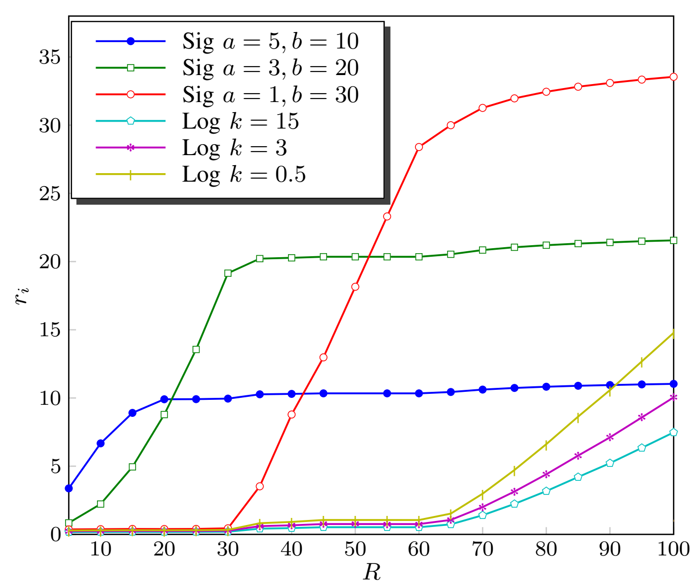

Fig. 8. The allocated rates r_i for different values of R and $\delta = 10^{-3}$ for Algorithm in (3) and (4).

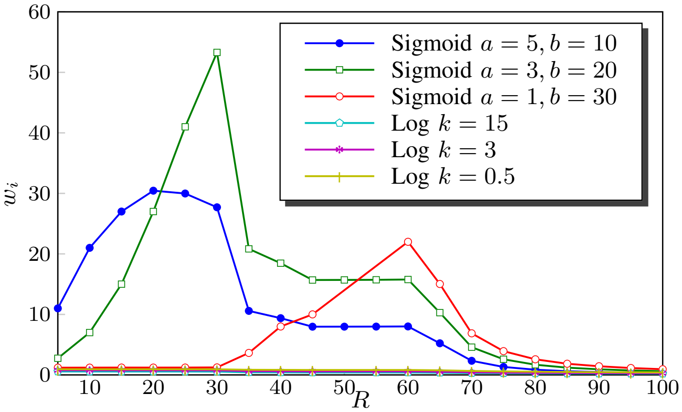

Fig. 9. The final bids w_i for different values of R and $\delta = 10^{-3}$ for Algorithm in (3) and (4).

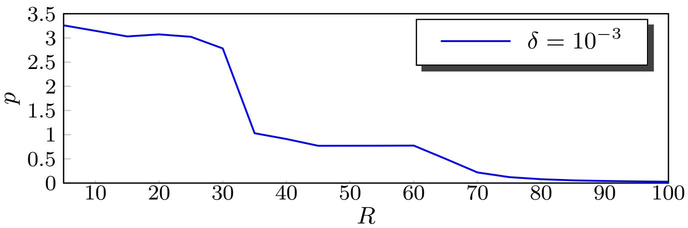

Fig. 10. The final shadow price p for different values of R and $\delta = 10^{-3}$ for Algorithm in (3) and (4).

rates $r_i(n)$ of different users with the number of iterations n for Algorithm in (1) and (2). It is shown that the rates fluctuate around the optimal rates and so the optimal rates is not achieved and the exit condition is not satisfied (i.e. endless iterations). Similar behavior for bids $w_i(n)$ with the number of iterations n is shown in Figure 5. Algorithm in (3) and (4) behavior is more robust due to the fluctuation decay function. It damps the fluctuation with every iteration so the network reaches the optimal rates of the optimization problem (3). The rates $r_i(n)$ and bids $w_i(n)$ of Algorithm in (3) and (4) are shown in Figures 4 and 6, respectively. Figure 7 shows the fluctuating shadow price $p(n)$ of Algorithm in (1) and (2) and the damping shadow price $p(n)$ of Algorithm in (3) and (4).

B. Rate Allocation and Pricing for $5 \leq R \leq 100$

In the following simulations, we set $\delta = 10^{-3}$ and the total rate of the eNodeB R takes values between 5 and 100 with step of 5. In Figure 8, we show the final rates of different users with different eNodeB rate R. Our distributed algorithm is set to avoid the situation of allocating zero rate to any user (i.e. no user is dropped). However, the eNodeB allocates the majority of the resources to the UEs running adaptive real-time applications until they reach the inflection rate $r_i = b_i$. When the total rate R exceeds the sum of the inflection rates $\sum b_i$ of all the adaptive real-time applications, eNodeB allocates more resources to the UEs with delay-tolerant applications, as shown in Figure 8, when eNodeB rate exceeds $R = 65$. This behavior is similar to that in [6] but with including eNodeB rate $R < 60$ where the bandwidth resources are scarce with respect to the users utilities. In Figure 9, we show the final bids of different users with different eNodeB total rate R. The higher the user bids the higher the allocated rate. The real-time application users bid high when the resources are scarce and their bids decrease as R increases. Therefore, the pricing which is proportional to the bids is traffic-dependent. This gives the service providers the option to increase the service price for subscribers when the traffic load on the system is high. Therefore, service providers can motivate subscribers to use the network when the traffic load on the network is low as they pay less for the same service. The shadow price $p(n)$ represents the total price per unit bandwidth for all users. In Figure 10, we show the shadow price $p(n)$ with eNodeB rate R. The price is high for high-traffic (i.e. fixed number of users but less resources, R is small) which decreases for low-traffic (i.e. same number of users but more resources, R is large). A large decrease in the price is apparent after $R = \{10, 30, 60\}$ which are the points where one of the users utility exceed the inflection point. This large decrease occurs at the sum of inflection points $\sum_{i=1}^{k} r_i^{\text{inf}}$, where $k = \{1, 2, ..., M\}$ is the users index and M is the number of users.

VI. Conclusion

In this paper, we presented the convergence analysis of resource allocation problem of hybrid traffic in 4G-LTE. We showed that prior methods are not convergent for different network traffic conditions. We proposed a robust optimal algorithm that converges for high-traffic and low-traffic loads occurring during the day. Our robust algorithm damps the fluctuation that could occur in low-traffic situations and converges to the optimal rates. In addition, we illustrated that our algorithm provides a pricing approach for network providers that could be used to flatten the traffic loads during peak traffic hours. We showed that our algorithm provides a traffic-dependent pricing that could be used by subscribers to decrease the cost of using the network by choosing to use the network at low-cost low-traffic periods.

References

[1] F. Kelly, A. Maulloo, and D. Tan, "Rate control in communication networks: shadow prices, proportional fairness and stability," in *Journal of the Operational Research Society*, vol. 49, 1998.
[2] S. H. Low and D. E. Lapsley, "Optimization flow control, i: Basic algorithm and convergence," *IEEE/ACM Transactions on Networking*, vol. 7, no. 6, pp. 861–874, 1999.
[3] J.-W. Lee, R. R. Mazumdar, and N. B. Shroff, "Downlink power allocation for multi-class wireless systems," *IEEE/ACM Trans. Netw.*, vol. 13, pp. 854–867, Aug. 2005.
[4] R. L. Kurrle, "Resource Allocation for Smart Phones in 4G LTE Advanced Carrier Aggregation," Master Thesis, Virginia Tech, Nov. 2012.
[5] S. Shenker, "Fundamental design issues for the future internet," *IEEE Journal on Selected Areas in Communications*, vol. 13, pp. 1176–1188, 1995.
[6] A. Abdel-Hadi and C. Clancy, "A Utility Proportional Fairness Approach for Resource Allocation in 4G-LTE," in *ICNC (submitted)*, 2013.
[7] G. Tychogiorgos, A. Gkelias, and K. K. Leung, "A new distributed optimization framework for hybrid ad-hoc networks," in *GLOBECOM Workshops*, pp. 293–297, 2011.
[8] G. Tychogiorgos, A. Gkelias, and K. K. Leung, "Towards a fair non-convex resource allocation in wireless networks," in *PIMRC*, pp. 36–40, 2011.
[9] T. Harks, "Utility proportional fair bandwidth allocation: An optimization oriented approach," in *QoS-IP*, pp. 61–74, 2005.
[10] G. Tychogiorgos, A. Gkelias, and K. K. Leung, "Utility-proportional fairness in wireless networks.," in *PIMRC*, pp. 839–844, IEEE, 2012.