Congestion Detection for QoS-enabled Wireless Networks and its Potential Applications

Ramneek Sekhon, Patrick Hosein, Wonjun Choi, and Woojin Seok

Abstract: **We propose a mechanism for monitoring load in quality of service (QoS)-enabled wireless networks and show how it can be used for network management as well as for dynamic pricing. Mobile network traffic, especially video, has grown exponentially over the last few years and it is anticipated that this trend will continue into the future. Driving factors include the availability of new affordable, smart devices, such as smart-phones and tablets, together with the expectation of high quality user experience for video as one would obtain at home. Although new technologies such as long term evolution (LTE) are expected to help satisfy this demand, the fact is that several other mechanisms will be needed to manage overload and congestion in the network. Therefore, the efficient management of the expected huge data traffic demands is critical if operators are to maintain acceptable service quality while making a profit. In the current work, we address this issue by first investigating how the network load can be accurately monitored and then we show how this load metric can then be used to provide creative pricing plans. In addition, we describe its applications to features like traffic offloading and user satisfaction tracking.**

Index Terms: **Cellular networks, congestion, long term evolution (LTE), mobile data offloading, pricing, quality of service (QoS), scheduling, wireless networks.**

I. INTRODUCTION

OVER the last few years, there has been a tremendous increase in mobile data traffic. According to statistics from Cisco's visual networking index 2014–2019 [1], the global mobile data traffic grew from 1.5 exabytes per month in 2013 to almost 2.5 exabytes by the end of 2014 and the monthly data traffic is expected to surpass 24.3 exabytes by the end of 2019. The growth has been exponential due to a number of contributing factors. Firstly, there has been a huge increase in the number of mobile data users and the number of smart connected devices on the network. As indicated in the traffic and market report by Ericsson [2], the number of mobile-connected devices exceeded the world's population in 2013. Nearly half a billion additional mobile devices and connections were added in 2014 [1]. Secondly, these mobile devices are becoming more powerful and thus are able to consume and generate more data traffic. Smartphones account for nearly 88 percent of the total growth in mobile data traffic due to the availability of data intensive applications and their ease of use. Finally, the availability of high capacity intelligent networks has further led to the widespread adoption of smart devices. In short, the rapid increase in the number of mobile data devices, subscriptions and the data intensive applications supported over them has led to a continuous increase in data traffic as well as average data volume per subscription [3]. Video traffic accounts for a majority of the traffic increase and it is anticipated that almost three quarters of mobile data traffic will be video by the end of 2019. This is of concern because interactive video traffic is resource intensive (high bit rate with delay constraints).

In order to meet this quickly growing demand, operators and network planners have focused on various channel allocation schemes and spectrum efficiency improvements in order to allocate resources more efficiently to users [3]. Some of the solutions include building new cell sites, reusing spectrum, offloading data onto other networks such as Wi-Fi and using femtocells. However each one of these has its own limitations. Although it is expected that carriers and service providers can manage capacity for a couple of years, the demand is inevitably going to exceed the available spectrum, leading to congested and overloaded networks. This can lead to an increased number of dropped calls, reduced data rates and increased prices and hence lower user satisfaction.

Pricing is now considered as a new means to control congestion [4]. The data plan (price and allowance) affects user behavior as the users are inherently price-sensitive. Low prices drive up the number of users and a larger data allowance typically drives higher volume per user. Pricing thus becomes a means for traffic management and congestion control. Naturally, there is a delicate balance between pricing and user performance (i.e., improved performance for a user should come at a cost). There must also be a balance between operator revenue (to fully utilize the resources) and customer satisfaction (excess capacity to allow for load fluctuations) and this is achieved through proper congestion detection and management as well as admission control algorithms. The challenge in providing support for the expected rapid growth in video applications can therefore be addressed by proper load monitoring and adjustable pricing schemes, which is what we propose in this paper. Proper load monitoring is critical for the network performance as congestion can significantly degrade performance and user satisfaction.

Manuscript received February 24, 2015; approved for publication by Yong Li, Division III Editor, July 27, 2015.

This work has been supported by the research project: Collaboration Platform Service Technology Development and Application (K-16-L01-C04-S03) of Korea Institute of Science and Technology Information (KISTI), Daejeon, Korea.

Ramneek is with the Department of Grid and Supercomputing, Korea University of Science and Technology, Daejeon, Korea and Department of Advanced KREONET Application Support, Korea Institute of Science and Technology Information, Daejeon, Korea, email: ramneek@kisti.re.kr.

P. Hosein is with the Department of Computer Science, the University of the West Indies, Trinidad and Tobago, email: patrick.hosein@sta.uwi.edu.

W. Choi is with the Department of Grid and Supercomputing, Korea University of Science and Technology, Daejeon, Korea and Department of Advanced KREONET Application Support, Korea Institute of Science and Technology Information, Daejeon, Korea, email: cwj@ust.ac.kr.

W. Seok is the corresponding author and is with the Department of Advanced KREONET Application Support, Korea Institute of Science and Technology Information, Daejeon, Korea, email: wjseok@kisti.re.kr.

Digital object identifier 10.1109/JCN.2016.000066

The contribution of this paper is twofold. First, we provide a load detection mechanism to monitor the load on the network. Secondly, we propose some applications of this load metric, such as a simple, flexible pricing scheme for quality of service (QoS)-based services. This scheme is beneficial to users as it allows them to dynamically request and pay for improved network performance whenever they need it. The proper monitoring of the network load ensures that additional users, additional flows per user and enhanced performance guarantees can be provided without significant degradation of service to those users already being served. In addition, we discuss the application of the above metric in triggering mobile data offloading to Wi-Fi networks or femtocells and in the monitoring of user satisfaction.

The paper has been organized as follows: Section I describes the problem being addressed; Section II provides an overview of QoS scheduling and the details of the proposed congestion detection mechanism, Section III includes the pricing model with illustrative examples, Section IV describes other applications of the proposed metric and Section V contains simulation results which illustrate the effectiveness of the approach.

II. CONGESTION DETECTION

In this section we provide the details of the load monitoring and congestion detection approach. Since it is based on information from the scheduler, we will first provide an overview of the QoS based scheduling.

A. An Overview of QoS-based Scheduling

Several schedulers have been proposed for scheduling users over a shared packet data channel. Some of them support QoS guarantees, e.g., [4], while others only maintain some degree of relative fairness. Long term evolution (LTE) provides a QoS framework and so we focus on how a generic QoS based scheduler works. First we provide a mathematical formulation of the scheduling problem and show that the optimal solution can be obtained using a gradient ascent method. We then introduce the notion of barrier functions that we use to enforce the QoS demands. The formulation of this shared resource allocation problem, as provided by Kelly [5], is as follows:

$$\text{maximize} \quad F(\vec{r}) \equiv \sum_{i=1}^{k} U_i(r_i) \tag{1}$$

$$\text{subject to} \quad \sum_{i=1}^{k} r_i < C \tag{2}$$

$$\text{over} \quad r_i \geq 0, \ \ 1 \leq i \leq k \tag{3}$$

where
k = the number of active users competing for the channel,
r_i = the average throughput of user i,
C = the channel capacity,
$U_i(r_i)$ = the utility function of user i.

In this formulation, the utility function has been defined in terms of user throughput but similar formulations are also possible for other QoS metrics such as delay. If we assume the utility function to be strictly concave and differentiable, then the same also holds for the objective function F. Since the feasible region is compact, a unique optimal solution exists, and can be found by Lagrangian methods. The optimal set of rates, which we denote by $\{r_i^*\}$, has the property that $U_i'(r_i^*)$ is the same for all users. Note that this optimal point can be computed explicitly. Assume that (exponentially smoothed) user throughputs are computed as follow

$$r_i(n+1) = \begin{cases} (1-\frac{1}{\tau})r_i(n) + \frac{\mu_i(n)}{\tau}, & \text{if } i \text{ served in slot } n \\ (1-\frac{1}{\tau})r_i(n), & \text{otherwise} \end{cases} \tag{4}$$

where $\mu_i(n)$ is the bit rate of user i if served during the nth period, τ is the time constant of the smoothing filter and $r_i(n)$ denotes the average throughput that was previously computed. If $U_i'(r_i(n))$ denotes the gradient of the utility function of user i at the start of the nth frame and let $F_j'(\vec{r}(n))$ denote the gradient of the objective function in the direction of serving user j, then the standard dual-ascent algorithm can be used to determine the value of j with the largest gradient and move to the maximal point along that direction. However in this case such arbitrary changes are not possible. If a user is chosen then that user is served and the throughputs are updated. This determines the amount of movement in the direction corresponding to serving the jth user. Therefore, the dual-ascent algorithm reduces to finding the maximum gradient direction and serving the corresponding user. One can show that the maximum gradient direction (i.e. user to be served) is given by

$$j* = \arg\max_j \{F_j'(\vec{r}(n))\} = \arg\max_j \{\mu_j(n) U_j'(r_j(n))\}. \tag{5}$$

Assume that the provider charges the users based on the number of bits delivered to the them. In this case, the utility is a linear function of the average throughout of a user and hence $U(r) = \alpha r$ for some constant α. The resulting scheduler will pick the user for which:

$$j* = \arg\max_j \{\mu_j(n)\} \tag{6}$$

which essentially means picking the user with the highest achievable bit rate. This results in maximum sector throughput and hence maximum revenue.

If the well known proportional fair utility function is used then $U(r) = \log(r)$ and hence the resulting scheduler picks the user such that:

$$j* = \arg\max_j \left\{ \frac{\mu_j(n)}{r_j(n)} \right\}. \tag{7}$$

This provides some degree of fairness at the expense of reduced sector throughput (since users in bad conditions must also be served). The utility function can be designed to provide QoS guarantees. This can typically be achieved by including with it some "barrier" function that penalizes movement into areas where the QoS requirement is violated. For example, if a lower bound on throughput is required then an appropriate barrier function can be chosen that will result in a rapid decrease in utility as the throughput approaches the throughput threshold [6].

B. Load Metric for Congestion Detection

The main reason for providing this review of the scheduler is to point out how it contains information that can be used to determine the congestion level in the network. We will refer to the gradient defined in the previous section as the priority function and hence the user with the largest priority function value is served at each scheduling decision. Note that in 4G networks, such as LTE, multiple users are served in a frame and so in this case these "scheduling decisions" are made multiple times for determining the users to be served in a frame. For our purposes we can ignore these details.

When a user is served (allocated more resources), its gradient (priority function) decreases and when it is not served, its gradient increases. The net result is that all users approach some common gradient (the optimal point for convex optimization). This common gradient changes as the load on the system changes. If the load is high (and users receive few resources), then the common gradient is also high. Hence, by monitoring the average of the gradients of the served users over time, one can monitor the load on the network. Note that this approach holds whether the utility is a function of throughput and/or delay. This metric (average priority of served users) will be used to detect the congestion level and to determine whether or not users can be "boosted" to achieve increased QoS guarantees. As discussed earlier, the user to be served is given by

$$j* = \arg\max_j \{\mu_j(n) U_j'(r_j(n))\} \tag{8}$$

and hence if we denote

$$p(n) = \mu_{j^*}(n) U_{j^*}'(r_{j^*}(n)) \tag{9}$$

as the priority of the served user and take a geometrically filtered average of this metric as

$$\bar{p}(n+1) = \epsilon\bar{p}(n) + (1-\epsilon)p(n) \tag{10}$$

for some filter constant $0 < \epsilon < 1$ then the metric $\bar{p}(n)$ can be used as a measure of the load at time instant n. This filter constant ε determines how much memory (or degree of smoothing) is included in the estimation of the average priority. Typically, a value of $\varepsilon = 0.99$ will provide sufficient memory.

To illustrate how this metric can be used, let us consider a simple scenario. Suppose that we use the proportional fair utility function for scheduling users and, in addition, we also wish to ensure that a minimum throughput r_{min} is maintained for all the users. Let μ_{min} represent the minimum rate that can be achieved by a user. This is the rate that would be achieved by a user at the edge of the cell. We assume that users with a signal to interference and noise ratio (SINR) smaller than that needed to achieve this rate are no longer able to maintain a connection. As the loading increases (due to either more users in the cell or more users in bad radio conditions), we find that $\bar{p}$ will increase. When it reaches a value of $\bar{p} = \mu_{min}/r_{min}$, any further increase in loading will cause the throughput of the edge user to drop below r_{min}. Hence, if new users are blocked when $\bar{p}$ reaches this threshold we maintain acceptable throughput levels for all users. A similar approach applies when we consider QoS based services as well.

To further illustrate the novelty of the proposed metric for load detection, consider the following examples. Consider the case of identical users with the same channel conditions. If the number of users is slowly increased then the frequency at which each one of them is served decreases (as more users have to be served). This leads to a decrease in their throughput. Hence, the priority function for that user increases (because the priority function (gradient) decreases as the user throughput increases). Since this will happen with all the users, the priority function for all the users will increase, which will result in a higher overall priority. Hence, the proposed metric (gradient or the average priority) reflects the congestion in the system. Consider another scenario with a fixed number of users. If all the users move from the center of the cell to the edge their signal strengths will decrease which will result in lower SINR and decrease in throughput for each one of them. Hence, similar to the above scenario, this will lead to increased priorities for all users and hence an increased average priority indicating an increased network load. Note that this metric works whether the utility is a function of throughput and/or delay.

C. Related Work

Congestion detection is a fundamental issue in networking and it is even more important for networks that are capable of provisioning QoS, as the QoS guarantees provided to the end users must be continued as the load on the network increases. Most of the existing end to end congestion control mechanisms are based on transmission control protocol (TCP) or its variants (e.g., [7], [8]). However, the TCP congestion control mechanism was initially designed for wired networks where it was assumed that the packet loss occurs because of buffer overflows at the intermediate nodes. Such assumptions are true for the wired networks in which the bandwidth-delay product is low. However, in the case of wireless technologies, such as LTE networks, such assumptions are no longer valid as there are a number of factors (such as handover, interference, noise, multi-path fading, etc.) other than congestion that can cause packet loss in the network.

Several congestion control mechanisms have been proposed for LTE systems and heterogeneous wireless networks. For instance, the congestion management approach in [9] is based on the media-independent handover protocol (MIH) where the congestion state, obtained from the radio resource control layer, is used in making vertical handover decisions. However, they mainly focus on node mobility and the decision is based on the network capacity and the number of available resources. Also, the application priority or type of traffic is not taken into consideration. In addition, [10] studies the combination of various queue-aware scheduling algorithms and congestion control mechanisms. In case of [11], the total load on the network is defined as the sum of the load on each bearer, and is compared to the threshold value for congestion detection. If the load remains above the threshold, for some stipulated time period a subset of bearers are dropped until the load reduces to an acceptable value. In this case removing the low priority bearers every time can lead to fairness issues and it is possible for the cell to become overloaded before congestion is detected.

Some of the existing congestion control mechanisms are based on parameters such as internal queue length, service time, inter-arrival time, and power variance. Such parameters have high processing overhead as they are dependent on the processing at the intermediate nodes. In addition, for video and streaming applications, parameters such as energy variance might have no impact. Another set of algorithms are based on measuring performance metrics such as throughput, latency, jitter, queuing delay, packet loss rate, etc. over time to indicate congestion in the network. However, it is challenging to measure such parameters as queuing delay reliably and such techniques may result in false detection in a number of scenarios. Additional examples are (a) congestion detection based on number of users (this technique was good enough for voice where the bandwidth per user did not vary much but it is not feasible for data intensive applications since one user can consume a large percentage of resources), (b) monitoring the total throughput (this technique works well if all the connections are throughput intensive, for e.g., web browsing but not if there are delay sensitive applications such as voice over Internet protocol (VoIP)).

The proposed algorithm is a low cost solution for congestion detection as it is based on measurement of the average priority over time and is not dependent on intermediate nodes for this information. It works well in wireless scenarios and is more robust than the existing techniques as it takes the QoS into consideration.

III. DYNAMIC QOS-BASED PRICING MODEL

In general, a good pricing model must be (a) simple, so that customers can easily understand how they are being charged, (b) fair, so that those who require more resources should pay accordingly and (c) efficient, so that the cost and complexity of implementing the scheme is minimal ([12], [13]). The current pricing plans are typically based on limits on data usage. The increased data allowance per month results in higher cost of the associated plan. Once a user reaches their data limit they must pay for additional data at a higher cost per bit than for their subscription plan. Note, however, there is no consideration of QoS in this case and users are provided with best effort services. With the advent of LTE and WiMAX, operators can now provide QoS based services. With QoS, a simple data usage based model is no longer valid as, for example, an application such as VoIP may require significant resources but the number of bits transmitted is small. This is due to the fact that VoIP is delay sensitive and in order to support low latency the allocation of resources is less efficient. Firstly, the payload of each VoIP packet is small and hence the overhead incurred is large. Secondly, when a VoIP packet arrives for transmission over the air, it cannot be queued excessively. This means that it may have to be transmitted even if the radio conditions are not favorable. In contrast, data packets can be queued until radio conditions are favorable and hence more efficient transmissions are achieved.

A. Proposed Pricing Strategy

Once QoS is implemented, the users must then pay according to how tightly their QoS guarantees are provided. Better QoS requires more allocated resources, which in turn should be reflected in higher pricing. Resources (bandwidth, power, time etc.) are allocated through the scheduler at the enhanced node B (eNodeB) for both uplink and downlink transmissions. In the case of LTE, bandwidth is allocated in resource blocks (RBs). In the downlink, power is typically allocated uniformly across all RBs and in the uplink, power is allocated based on the uplink power control algorithm. Therefore, we focus on the scheduling of RBs across users.

According to the proposed pricing strategy, the subscription plans will be classified in terms of the QoS requirements of different user applications.The users with high QoS requirements must subscribe to higher service plans. The higher subscription plans will guarantee a higher degree of QoS resulting in higher pricing as compared to the lower service plans. The scheduler prioritizes users and allocates resources based on this ordering. As discussed in the previous section, this priority can be a function of throughput and/or delay depending on which QoS metric is of concern for the flow. Note that if a particular service requires constraints on both throughput and delay, then typically one of them will be dominant and so resources can be allocated based on the more stringent QoS requirement. Packet error rate is handled by the hybrid ARQ re-transmission mechanism and jitter is not covered in LTE.

As discussed above, over time all users will approach a common priority level (which we call $\bar{p}$). Consider a throughput based priority function such as that used in the proportional fair scheduler. If we denote the achievable rate of mobile i by μ_i (assumed nearly constant over the convergence period) and the achieved throughput by r_i then the associated priority level is given by μ_i/r_i and in steady state we have

$$\bar{p} = \frac{\mu_i}{r_i}. \tag{11}$$

Now suppose that we apply some weight κ_i to the priority of this mobile while performing the scheduling. The value κ_i will depend on the service plan that the user is subscribed for. The limiting throughput as a function of this weighting is given by

$$r_i = \frac{\kappa_i \mu_i}{\bar{p}}. \tag{12}$$

Note that in reality, the limiting priority $\bar{p}$ will change as κ_i changes but for a sufficiently large number of mobiles (which is the case in 4G networks) this dependence will be small. We therefore find that the achievable rate of the user can be increased by increasing its weighting κ_i.

Let us look at the case of a delay sensitive application such as VoIP. If the present delay for the flow is d_i and the required maximum delay is d_{max} then a typical priority function is

$$p_i = \mu_i \left(1 - \frac{d_i}{d_{max}}\right)^{-1}. \tag{13}$$

In this case the metric of concern is d_i and the lower this value the better the quality of the call. Note that the second component on the right is what we have been calling a barrier function as its aim is to prevent the delay from exceeding its maximum value. Again, if we apply some weighting κ_i and assume that the limiting priority $\bar{p}$ has been achieved, then we have

$$d_i = d_{max} \left(1 - \frac{\kappa_i \mu_i}{\bar{p}}\right) \tag{14}$$

and again we find that as the priority weighting for the user is increased, its performance is improved through a lowering of d_i.

Hence, we can differentiate the degree of QoS conformance for different users by varying the weighting κ_i that is applied to their priority computed in the scheduler. This notion will be used to offer different pricing plans, as discussed in the next section, and works for any of the QoS metrics defined for 4G networks. By using different priority weights, we can provide differential performance treatment to different users.

The duration over which services are used must also be taken into account when charging. At present this is taken into account separately for voice and for data by having data caps for different data plans and having caps on voice minute usage for voice plans. In the present networks, data is typically carried over an LTE network but voice is still carried over 3G networks such as WCDMA or GSM and so this charging approach works well since delay sensitive voice (priced based on duration) is treated separately to delay tolerant data (priced based on transferred bits). Once operators start providing voice service over LTE (e.g., with VoLTE) then voice and data packets are no longer easily distinguishable. Furthermore, different constraints (and hence different quality of experience) may be applied to different versions of the same service. For example, one may choose a best effort flow for Skype service or a delay sensitive flow for VoLTE. Hence, charging for voice services will become more complicated.

Again note that the transmitted bits per resource for voice is much lower than for data, since the former has a constraint on delay. Let ρ_v denote the average transmission rate over a RB for voice transmissions and ρ_d be the rate for data transmissions. If r_v is the throughput for the voice application, then a good estimate for the data throughput *that could have been achieved* if the resources were used by the mobile for a delay tolerant application would be $(r_v \rho_d/\rho_v)$. One could then use this throughput estimate in computing the data usage for the user and in this way a single data cap is used for all services.

B. Sample Subscription Plan

On the basis of the arguments provided in the previous sections, we provide an example of a simple pricing plan. Consider the case of 3 user service classes namely Bronze, Silver, and Gold. By providing different priority weighting factors, we can provide different levels of service to each of these classes. For example, a priority weighting factor of $\kappa = 1$ can be used for Bronze users, $\kappa = 2$ for Silver users and $\kappa = 3$ for Gold users. Consider a Bronze and a Silver user. Suppose that both are running applications that require best effort service and that a proportional fair priority function is used for each flow. Furthermore, assume that they each have comparable achievable rates (i.e. radio conditions). In steady state we have

$$\bar{p} = \frac{2\mu}{r_{silver}} = \frac{1\mu}{r_{bronze}} \tag{15}$$

and so the ratio of the throughput of the Silver user to that of the Bronze user is simply the ratio of their priority weights 2. Hence, on average the Silver user will achieve twice the throughput than that of the Bronze user (with both in similar radio conditions).

Similarly, if they were both running the same delay sensitive application then the ratio of their average delays is given by

$$\bar{p} = \frac{d_{max}(1 - 2\mu/\bar{p})}{d_{max}(1 - 1\mu/\bar{p})}. \tag{16}$$

and so the Silver user experiences lower delays on average. For example, if we try to achieve an average delay of $2d_{max}/3$ for a bronze user for acceptable performance then the corresponding average delay for the silver user will be $d_{max}/3$ or half the delay of the bronze user. User subscriptions are priced proportionally to their priority weights κ and achieve performances comparable to the paid subscription rate.

C. Service Boost Provisioning

There are occasions when a user may desire better performance than provided by their base subscription plan. To address such a situation, we also propose a performance boost mechanism whereby a user can temporarily request an increase in service level (e.g., from Bronze to Silver or from Silver to Gold) in order to run an application with greater performance demands. The boost provisioning mechanism has been depicted in Fig. 1. When a user makes such a request, the weighting factor κ is determined according to the base subscription plan of the user. Using this value of κ, the priority is calculated. To determine whether the requested boost can be granted or not, the average priority is calculated and compared to some priority threshold. If the average priority is still less than the threshold then the boost is granted. However, if it is greater than or equal to the threshold, the boost request is denied. That is, if sufficient resources are available, the user is boosted to the higher level for the period of time requested. The user is then billed additionally based on the level to which they were boosted as well as the duration of time requested. Naturally the price rate for this boost will be higher than the rate charged for the basic subscription for the same class. For example, a normal user whose primary interest is web surfing and email may initially subscribe to a Bronze plan. If at any point of time they need to make a Skype call they can request a boost to the Gold priority weighting and specify the duration of this request. In order to determine whether the user can be boosted without significantly impacting other users, we must determine the present loading. This is where we use the average priority of served users $\bar{p}$ to make this decision. For example, if $\bar{p} < p_t$ then requested boosts can be accommodated otherwise they are refused. Naturally additional thresholds can be set and more fine tuned controls can be implemented.

D. Zero-Rated Applications

In addition to properly charging users for the resources they utilize, we can also take into account zero-rated applications. Zero-rated applications are provided free of charge by network operators. Such applications, also known as toll-free applications, are becoming popular in a number of countries. According to [14], nearly 45 percent of operators worldwide are currently providing at least one zero-rated application to their subscribers. Such applications are either data intensive involving streaming media or other core applications including text-based applications or social media (e.g., Facebook, Twitter, and Wikipedia).

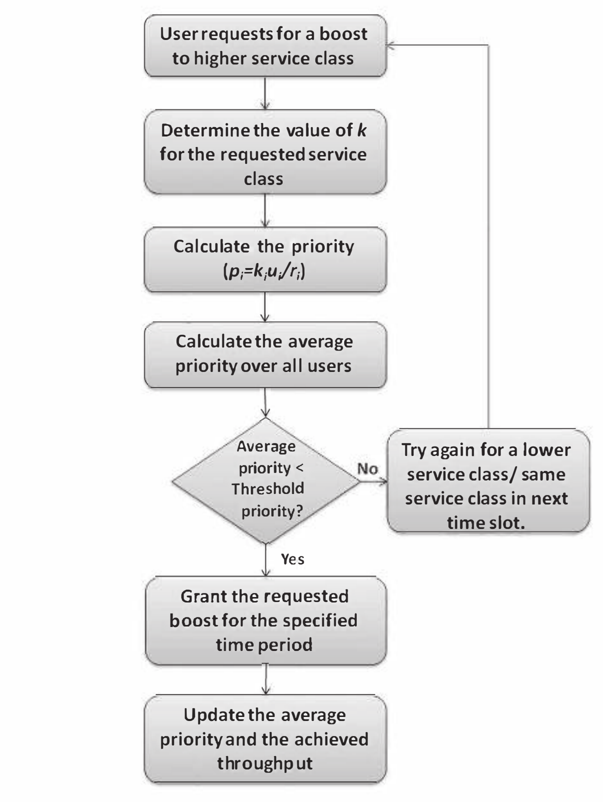
User requests for a boost to higher service class
Determine the value of k for the requested service class
Calculate the priority $(p_i=k_iu_i/r_i)$
Calculate the average priority over all users
Average priority < Threshold priority?
No
Try again for a lower service class/ same service class in next time slot.
Yes
Grant the requested boost for the specified time period
Update the average priority and the achieved throughput

Fig. 1. Boost request process.

The main motive behind providing zero-rated applications is to earn more revenue by attracting more users. In some cases the content provider pays the operator for providing the zero-rated application to its customers. Such data is commonly known as sponsored data [15]. In this way, the operator earns from the payment provided by the content provider and the content provider in turn earns revenue through indirect means such as advertisements.

Although the main motive behind providing such zero-rated applications is to increase the revenue for the ISP or the content provider, such data allowances or zero price schemes can increase the load on the system which can affect overall performance. In our proposed solution the operator can provide low priority to such applications while maintaining an acceptable service quality for the paid or premium applications even during high loads. The operator can fix upper and lower threshold values, denoted by p_{max} and p_{min} respectively. Once the congestion metric exceeds the upper threshold a very small weighting κ is applied to all zero-rated application bearers and once the average priority drops below p_{min} then this factor can be removed. Note that in the case of LTE networks [16] there can be more than one dedicated bearer attached to each user. In this way, by monitoring the load using the proposed congestion detection metric the operator can adjust the priority of zero-rated applications dynamically and thereby manage the load in the system.

E. Related Work

There are a number of static and dynamic pricing schemes that have been proposed in the literature [17]. In case of static pricing schemes (e.g., flat rate pricing, usage-based pricing, time-of-day pricing, etc.), prices change after a relatively long time period and so offered prices do not vary with respect to network congestion levels. Even though some pricing schemes such as Paris metro pricing [18], [19] support differentiated service classes, our proposed pricing method (i.e., applying scheduling weights based on subscription plan) provides a higher level of flexibility by allowing the users to request for a higher service class (boosting) only when the need arises. This not only helps in cost savings for the customer but also helps to control network congestion.

Another advantage of our proposal is reduced complexity. For example, in the case of the dynamic priority pricing scheme in [20], the user service requests are modeled as a stochastic process and network nodes as priority queues. Every incoming user request has an instantaneous value for the service and a corresponding linear rate of decay to capture the delay cost. This pricing model is based on achieving equilibrium which is rare in cellular networks due to fluctuating channel conditions. In contrast, the boosting decisions and the service class differentiation in our proposed pricing model depends on the congestion metric that we have defined in the previous sections, and it is relatively simpler to monitor and implement. In [21], a variant of usage-based pricing, users are charged based on a self-specified peak, mean traffic rates, the observed mean rate and the duration of each connection. While effective bandwidth pricing is fairly simple, it requires the users to know, or estimate, the peak and mean rates of each connection and an explicit bandwidth formula is required. In case of our proposed pricing plan, the user does not need to explicitly specify his requirements in terms of network parameters and can choose the service plan based on his general usage behavior. A corporate user, who frequently has to make Skype calls, may need to subscribe for a gold plan, while for the average user, whose primary interest is browsing and chatting, a bronze plan may be sufficient. The key point that makes our pricing strategy different from traditional QoS based pricing models is the flexibility to switch to a higher service plan on request.

IV. OTHER APPLICATIONS

A. Offloading Data from Cellular to Wi-Fi

Due to the ever increasing volume of mobile data traffic, as discussed in Section I, the available spectrum is getting exhausted. Although the operators can handle the capacity for a few more years the demand is ultimately going to exceed the available spectrum. One approach to alleviating network congestion and enhancing QoS is mobile data offloading. As projected by the Cisco VNI [1], by the end of 2016, more than half of mobile data traffic will be offloaded to Wi-Fi networks and femtocells. Benefits of mobile data offloading include better energy efficiency, reduced cost and performance gains achieved by the more efficient use of available resources. Offloading can either be initiated by the user for cost control or performance or by the operator to manage network congestion.

A number of algorithms have been proposed in the literature for efficient data offloading such as those in [22]–[25]. Decisions may be based on energy efficiency considerations for the mobile devices [26], or on machine learning based schedulers [27] or, in some cases, temporal techniques such as delayed offloading. In the latter case the data transfer has an associated delay deadline and the transfer can be resumed whenever the mobile node enters WiFi coverage. In addition, [28] focuses on delay tolerant networking systems and offloading decisions are based on whether the content to be offloaded falls within a specific category. Some other techniques such as offloading via opportunistic communication have also been proposed as low cost solutions where the content from the application providers is delivered to a small group of target users and this information can be relayed to other nodes in proximity using WiFi or bluetooth networks. However, such techniques are challenging due to the heterogeneous, time variant nature of user requirements [29].

In cases where offloading is initiated by the cellular provider our proposed metric can be used to trigger offloading. As described in Section II, the average priority over all users indicates the load on the system at any given point in time. Similar to the case of zero rated applications as described in Section III, the operator can define upper and lower threshold values as p_{max} and p_{min}, respectively. When the value of the load metric crosses the lower threshold (i.e. when $\bar{p} > p_{min}$) a congestion warning is generated and appropriate actions taken (e.g., by adjusting user scheduling weights). If congestion continues then when $\bar{p} > p_{max}$ data offloading should be triggered.

B. Monitoring User Satisfaction

Network performance monitoring is very important not only from the network management point of view but also from the user perspective. Service providers or network operators often use QoS based parameters such as throughput, delay, packet loss, jitter, etc., for assessing network quality. On the other hand, users are more concerned with the quality of experience (QoE) which is more subjective than QoS and is assessed in terms of non-technical parameters such as fast network connection, reasonable response time, etc. From the end user point of view, the technical network parameters might be unimportant and indistinguishable. There are a number of web-based network performance monitoring tools available to users [30]–[33]. These tools mainly rely on the measurement of parameters such as delay, throughput, etc. but they do not capture the user experience performance. The proposed load detection metric can provide the user with information regarding the level of service they are receiving at any particular instant of time. It provides the amount of resources available and hence reflects the user QoE. Whenever a user is served, or allocated some resources, its priority function will decrease and when it is not served, the priority increases. The net result is that all users approach some common value which changes as the load on the system changes. Hence, by monitoring the average of the priority of the served users over time, one can monitor the perceived user experience.

V. PERFORMANCE EVALUATION

In this section, we provide the framework used for our simulations followed by the results for some illustrative examples. First, we illustrate the effectiveness of the proposed load metric. We then illustrate the performance boosting feature described in Section III. Although the proposed metric is applicable to different wireless networks supporting QoS, the simulations were carried out for cellular networks. The simulations were performed using the LTE module of the network simulator-3 (NS-3) [34], which has been designed to support the evaluation of various aspects of LTE systems including radio resource management, QoS-aware packet scheduling, inter-cell interference coordination and dynamic spectrum access. The simulation parameters are summarized in Table 1. To illustrate the congestion metric, the the number of users and the target bit rate was varied dynamically, while for illustrating the service boosting, the number of users was fixed.

Table 1. Simulation parameters.

A. Congestion detection metric	
Number of UEs	4 to 160
Scheduling algorithm	PSS and PF
Target bit rate	250 Mbps (first use case) and variable (second use case)
B. Service boosting	
Number of UEs	50
Cell radius	50 meters
Common parameters	
Number of eNodeBs	1
Mobility model	Constant position mobility model
Propagation model	Friis spectrum propagation model
Simulation time	10 seconds

A. The Congestion Detection Metric

In order to investigate the feasibility of using average priority as a load metric in QoS-enabled wireless networks, the priority set scheduler (PSS), and a modified version of the traditional proportional fair (PF) scheduler of the LTE model of NS-3 [35] was used. The PSS scheduler was chosen because it is a QoS aware medium access control (MAC) scheduler and hence appropriate for this purpose. PSS is a combination of time domain and frequency domain packet schedulers and controls fairness among the UEs by the target bit rate (TBR). In PSS, the frequency domain scheduler allocates resource block group (RBG) k to the UE that has the maximum value of the chosen metric. This metric is calculated using the proportional fair (PFsch) or carrier over interference to average (CoIta) metric. The final priority metric is the product of the CoIta metric, which is dependent on the SINR value, and the weight as defined in [35]. For the second example a modified version of the traditional PF scheduler of the NS-3 LTE model was used. In this case, the utility function is defined in terms of the past throughput performance achieved by a user and its QCI class/priority as defined by the dedicated bearer between the UE and the eNodeB. In LTE networks QoS is implemented between the user equipment (UE) and packet data network (PDN) Gateway with the help of a set of bearers. In the case of the traditional PF algorithm, the priority metric is the ratio of the achievable rate (represented by the channel quality index) of the user and the average user throughput. This provides some degree of fairness but does not take into account the QoS requirements. We have modified the PF algorithm to include the QCI class information in the priority

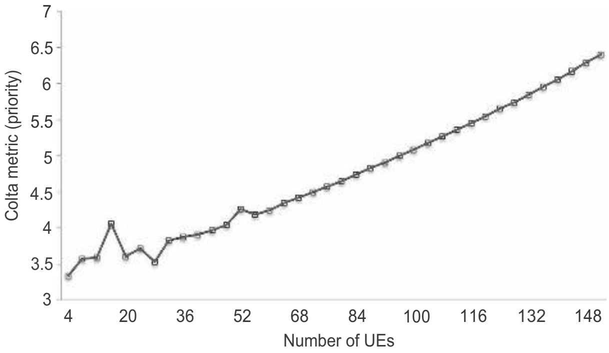

Fig. 2. Average priority vs. number of UEs for the PSS scheduler.

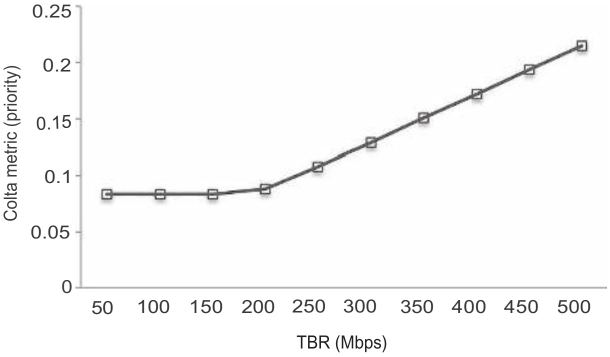

Fig. 4. Average priority vs. TBR for the PSS scheduler.

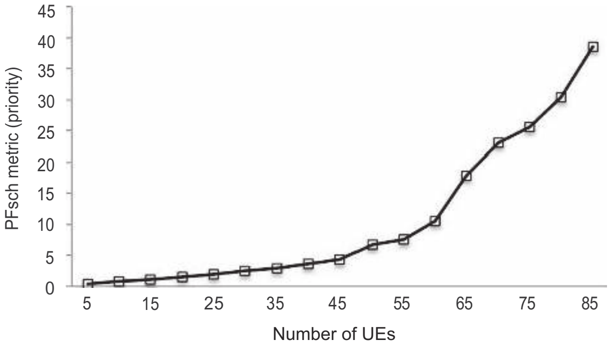

Fig. 3. Average priority vs. number of UEs for the PF scheduler.

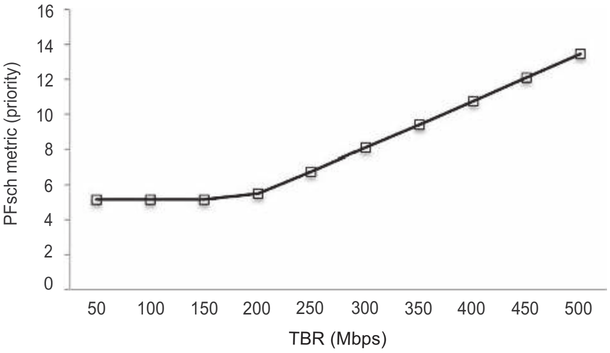

Fig. 5. Average priority vs. TBR for the PF scheduler.

calculation. This priority is defined as a function of throughput but it can also be defined in terms of delay depending on which QoS metric is of concern for the flow. The resource block is allocated to the user with the highest priority.

To simulate the load in the system, two use cases were taken into consideration for both schedulers (PSS and PF). For the first use case, the number of UEs was increased dynamically to increase the load on the system, and the priority metric (CoIta metric×weight for PSS scheduler) and ((achievable bit rate/past throughput)×weight for PF scheduler) was monitored. The results are shown in Fig. 2 and Fig. 3 for the PSS and PF schedulers, respectively. As the number of users increases, the system gets congested. The graphs clearly indicate that as the number of users increases the average priority increases in both the cases.

In the second use case the target bit rate (TBR) was increased while keeping the number of users constant. As the target bit rate increases the priority of the users will also increase as both schedulers attempt to satisfy the specified TBR to support the QoS. Hence the system load increases with increased TBR. The TBR was varied from 50 Mbps to 500 Mbps and the number of UEs was kept constant at 50. The results for both PSS and the modified PF scheduler are shown in Fig. 4 and Fig. 5, respectively. The system starts detecting heavy load at around 200 Mbps and thereafter the average priority increases almost linearly.

B. Service Boosting

In this section we illustrate the proposed on-demand service boosting feature. While the users can choose their base service plans depending on their general usage they may sometimes require a higher service quality for a given duration of time. We simulate a fixed number of UEs in a single cell connected to a single eNodeB. The simulation parameters are as defined in Table 1 (Part B), and the remaining parameters have been set to default values (see [34] for detailed modeling assumptions). As described earlier, the proportional fair MAC scheduler that was provided with the LTE module [35] was modified to include the priority weights κ that were described previously to differentiate users by subscription class. The weight κ, representing the subscription plan, depends on the radio bearer defined between the eNodeB and its attached UE. When a UE first connects to the network, it is assigned the default bearer. The default bearer provides best effort service. Dedicated bearers are then added to support different QoS applications. Although mobility is not included explicitly, it is accounted for in the fading model used in the simulation. Initially all mobile users are assigned bearers for a Bronze subscription service with a weighting factor of $\kappa = 1$. From time $t = 10$ to $t = 35$ s, one UE is boosted to the Silver subscription level with $\kappa = 2$. The UE is reduced to the Bronze class after this period.

The throughput and priority of a non-boosted user and the boosted user are provided in Fig. 6. The upper plot shows the variation of throughput for the two users. We can clearly see the improved performance of the boosted user. In this scenario, the non-boosted UE is in better radio conditions than the boosted UE and so the ratio of throughput is not 2 as in the case of our numerical example. Note the instantaneous rise in throughput of the boosted user before settling down to its steady state boosted

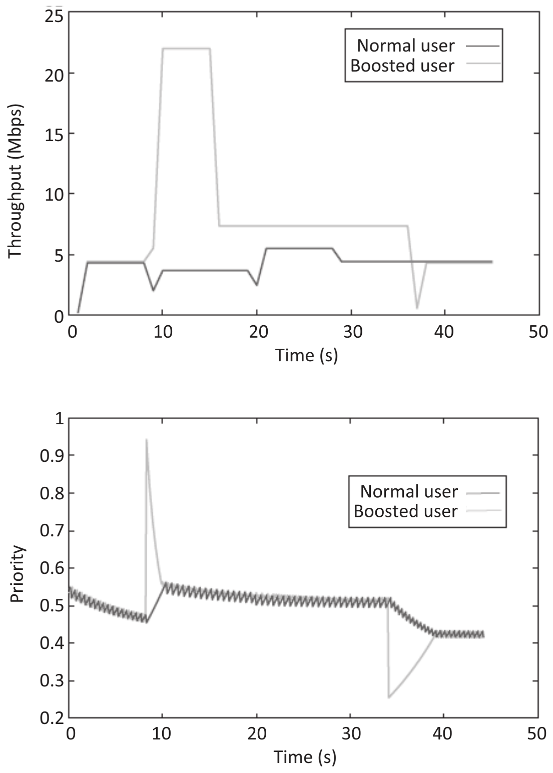

Fig. 6. Throughput and priority levels for boosted/non-boosted users.

throughput. The lower plot shows the variation of the priority values for the two users. In steady state, all users converge to a single priority level, as pointed out in the previous sections, so both boosted and non-boosted users achieve the same steady state priority value. However, we do see that when the boost is applied the average priority increases because of the increased loading of the system. Similarly, when the boost period ends, the priority level decreases because the user is switched back to the lower service class and the network load decreases. Therefore, the example clearly illustrates the effect that priority weights have on the performance of users.

VI. CONCLUSIONS AND FUTURE WORK

Due to an ever increasing growth in wireless data traffic the probability of network congestion has also increased. In order to properly manage available resources, accurate load detection and prediction methods are needed. This is even more important in networks where QoS is provided since one must ensure that those users who have been provided QoS guarantees continue to be provided with those guarantees as the load in the network increases. With increasing demands of wireless data users for both enhanced performance and more stringent QoS requirements, there is a need for wireless operators to better price their services to ensure that those who require improved performance can in fact achieve it but such users must, in return, pay for the improved service. To address these issues, we first proposed a mechanism for monitoring the load in a wireless network and, using this metric, we introduced a dynamic, flexible QoS-based pricing strategy. This congestion metric is based on information that is already computed by the QoS-based schedulers. In order to provide class based subscription services, we introduced weights, based on subscription class, that are applied to each user's priority value during the scheduling process. We also introduced the notion of service boosting whereby a user can request, and pay for, increased performance beyond what is provided by their paid subscription plan. In order to do this, one needs to ensure that sufficient resources are available to provide the boost while maintaining QoS guarantees to all other users. Again the proposed metric can be used for this purpose. We illustrated our proposed methods through simulations. Future work will entail simulations for a wider range of use cases and the introduction of additional pricing plans to accommodate the wide range of applications being offered in today's networks.

REFERENCES

[1] "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2014-2019," Cisco Whitepaper, www.cisco.com.

[2] "Traffic and market report, June 2012," www.ericsson.com.

[3] "Mobile broadband capacity constraints and the need for optimization," Rysavy Research, Feb. 2010.

[4] G. Song and Y. (G). Li, "Cross-layer optimization for OFDM wireless network: Part I and part II," *IEEE Trans. Wireless Commun.,* vol. 4, no. 2, pp. 614–634, Mar. 2005.

[5] F. Kelly, "Charging and rate control for elastic traffic," *European Trans. Telecommun.*, vol. 8, pp. 33–37, 1997.

[6] P. Hosein, "QoS control for WCDMA high speed packet data," in *Proc. IEEE WCNC,*, Stockholm, Sweden, Sept. 2002.

[7] H. Zhang, X. Zhang, B. Fan, and L. Shao, "Adaptive fast TCP," in *Proc. ICFN*, Jan. 2010, pp. 114–118.

[8] K. Xu, Y. Tian, and N. Ansari, "Improving TCP performance in integrated wireless communication networks," *Elsevier Comput. Netw.,* col. 47, pp. 219–237, 2005.

[9] H. Mzoughi, F. Zarai, M. S. Obaidat, and L. Kamoun, "3GPP LTE-advanced congestion control based on MIH protocol," *IEEE Syst. J.,* no. 99, pp. 1–11.

[10] A. Zolfaghari and H. Taheri, "Queue-aware scheduling and congestion control for LTE," in *Proc. ICON*, Dec. 2012, pp.131–136.

[11] R. Kwan, R. Arnott, R. Trivisonno, and M. Kubota, "On pre-emption and congestion control for LTE systems," in *Proc. IEEE VTC*, Sept. 2010.

[12] P. Hosein, "Pricing for QoS-based wireless data services and its impact on radio resource management," in *Proc. IEEE GLOBECOM Workshops,* Miami, USA, Dec. 2010.

[13] W. Premchaiswadi and S. Pattanavichai, "Pricing model and real options in 4G LTE mobile network," in *Proc. IEEE/ACIS SNPD,* Aug. 2012.

[14] Report: 45% of operators now offer at least one zero-rated app. [Online]. Available: http://www.fiercewireless.com, Feb. 2015.

[15] M. Andrews, G. Bruns, and H. Lee, "Calculating the benefits of sponsored data for an individual content provider," in *Proc. CISS*, Mar. 2014.

[16] Policy and charging control architecture (3GPP TS 23.203 version 12.7.0 Release 12). [Online]. Available: http://www.3gpp.org, Dec. 2014.

[17] S. Sen, C. J.-Wong, S. Ha, and M. Chiang, "A survey of smart data pricing: Past proposals, current plans, and future trends," *ACM Comput. Surv.*, vol. 46, no. 2, Article 15, Nov. 2013.

[18] A. Odlyzko, "Paris metro pricing for the Internet," in *Proc. ACM EC*, New York, NY, USA, pp. 140–147.

[19] S.Wahlmueller, P. Zwickl, and P. Reichl, "Pricing and regulating quality of experience," in *Proc. NGI*, June 2012.

[20] A. Gupta, D. Stahl, and A. Whinston, "Priority pricing of integrated services networks," *Internet Economics,* L. W. McKnight and J. P. Bailey, Eds. The MIT Press, Cambridge, MA, 323–352, 1997.

[21] F. Kelly, "On tariffs, policing, and admissions control for multiservice networks," *Oper. Res. Lett.,* vol. 15, pp. 1–9, 1994.

[22] F. Rebecchi *et el.,* "Data offloading techniques in cellular networks: A survey," *IEEE Commun. Surveys & Tutorials*, vol. 17, no. 2, pp. 580–603.

[23] K. Lee, J. Lee, Y. Yi, I. Rhee, and S. Chong, "Mobile data offloading: How much can WiFi deliver?," *IEEE/ACM Trans. Netw.*, vol. 21, no. 2, pp. 536–550, 2013.

[24] M. H. Cheung and J. Huang, "Optimal delayed Wi-Fi offloading," in *Proc. WiOpt*, May 2013, pp. 564–571.

[25] L. Gao, G. Iosifidis, J. Huang, L. Tassiulas, and D. Li, “Bargaining-based mobile data offloading,” *IEEE J. Sel. Areas Commun.,* vol. 32, no. 6, pp. 1114–1125, 2014.

[26] A.Y. Ding *et al.,* “Enabling energy-aware collaborative mobile data offloading for smartphones,” in *Proc. IEEE SECON*, June 2013, pp. 487–495.

[27] H. Eom *et al.,* “Machine learning-based runtime scheduler for mobile offloading framework,” in *Proc. IEEE/ACM UCC,* Dec. 2013, pp. 17–25.

[28] Y. Li *et al.,* “Multiple mobile data offloading through delay tolerant networks,” in *Proc. ACM CHANTS*, 2011, pp. 43–48.

[29] A. Aijaz, H. Aghvami, and M. Amani, “A survey on mobile data offloading: Technical and business perspectives,” *IEEE Wireless Commun.*, vol. 20, no. 2, pp. 104–112, 2013.

[30] [Online]. Available: http://www.speedtest.net

[31] C. Kreibich, N. Weaver, B. Nechaev, and V. Paxson, “Netalyzr: Illuminating the edge network,” in *Proc. ACM IMC,* 2010, pp 246–259.

[32] [Online]. Available: http://www.broadbandspeedchecker.co.uk/

[33] [Online]. Available: http://www.cnet.com/internet-speed-test/

[34] [Online]. Available: http://www.nsnam.org/docs/release/3.10/manual/html/lte.html

[35] [Online]. Available: http://www.nsnam.org/docs/release/3.18/models/ singlehtml/index.html# document-lte

Ramneek Sekhon received B.Tech. (Computer Science and Engineering) and M.Tech. (Information Technology) from Guru Nanak Dev University (G.N.D.U) Amritsar, India in 2010 and from International Institute of Information Technology (IIITB), Bangalore, India in 2013, respectively. She is currently a Ph.D. student of the University of Science and Technology (UST), Daejeon, Korea, at Korea Institute of Science and Technology Information (KISTI). Her research interests include networking and communication (congestion management, QoS and pricing, admission control, resource management, network neutrality) and cloud computing (openstack, IaaS, cloud federation, cloud networking, SDN).

Patrick Hosein attended the Massachusetts Institute of Technology (MIT) where he obtained five degrees, a B.Sc. degree in Electrical Engineering and one in Mathematics, an M.Sc. degree in Electrical Engineering and Computer Science, an Engineer’s degree, and a Ph.D. in Electrical Engineering and Computer Science. He has worked at Bose Corporation, Bell Laboratories, AT&T Laboratories, Ericsson, and Huawei. He is presently a Professor of Computer Science at the University of the West Indies, St. Augustine, Trinidad. He has published extensively with over 75 refreed journal and conference publications. He holds 38 granted and 42 pending patents in the areas of telecommunications and wireless technologies. His present areas of research include radio resource management, QoS, and Pricing for 5G cellular networks.

Wonjun Choi received his Bachelor’s degree in Mathematics from Wonkwang University in 2006. He is currently enrolled in the integrated course in Grid and Super computing in the Korea University of Science and Technology. His research interests include network engineering, network communication, TCP, cloud federation, CCNx, NDN, SDN, and computer science.

Woojin Seok received B.E, M.S, and Ph.D. from Kyungpook National University, University of North Carolina at Chapel Hill, and Chungnam National Univerysity, respectively. He currently works for Korea Institute of Science and Technology Information (KISTI) Advanced KREONET center. He is also Adjunct Professor of University of Science and Technology (UST). He is a Society Member of KICS and KIPS, and also committee member of Future Internet Forum (FIF) Korea and Asia Pacific Network Operation and Management (APNOM). His interesting research areas are network testbed, federation, SDN, and so on.