User Experience Aware Active Queue Management in Cellular Networks

Yuhang Dai, Vindya Wijeratne, Yue Chen, and John Schormans
School of Electronic Engineering and Computer Science.
Queen Mary University of London
London, UK
Email: {y.dai, vindya.wijeratne, yue.chen, j.schormans}@qmul.ac.uk

***Abstract*—**Active queue management plays an important role in the performance of all IP based networks and cellular networks in particular where large buffers are deployed in order to absorb bursts resulting from the dynamic nature of the radio channel. Long standing queues building up in large buffers leads to "Bufferbloat", degrading the performance especially for delay-sensitive applications while small buffers may lower the link utilization. This paper proposes a novel AQM algorithm tailored to cellular networks, mainly by utilizing the Quality of Experience (QoE) metric in order to mitigate Bufferbloat and maintain acceptable levels of performance. Simulation results show the proposed algorithm provides a good balance between drops and delay hence successfully maintains expected levels of service.

***Index Terms*—**QoE, AQM, cellular networks

I. Introduction

The emergence of powerful smart devices and their integration in people's daily lives place huge strains on networks. As forecast by Cisco [1], global mobile traffic will increase eight-fold by 2020 and the link speed will increase three-fold. Since 4G, cellular networks have been all IP-based and applications such as video streaming, gaming and online chatting are widely used. Large buffers are often deployed in intermediate devices in order to absorb bursty traffic and increase link utilization. Over-buffering results in long standing queues which leads to huge queuing delays as well as a reduction in overall network throughput, which is referred to as the "Bufferbloat" phenomenon [2].

[3] has demonstrated the Bufferbloat issue in broadband networks and [2] shows that the issue exists in both uplink and downlink directions. Bufferbloat is also reported in wireless access networks, as shown in [4] (Wi-Fi) and [5] (cellular networks). Wireless devices are often equipped with large buffers due to time-varying wireless channels, which makes Bufferbloat potentially severe in cellular networks.

Solutions to alleviate Bufferbloat can be divided into 2 categories. One is modifications to TCP on the realization that over buffering delays the signaling of congestion and the other one is Active Queue Management (AQM) where packets are actively dropped prior to reaching the full-buffer status.

Variations of TCP have been proposed which try to sense the congestion as early as possible, such as TCP Vegas [6] which is delay-based; however, it faces fairness issues when competing with loss-based TCP. TCP Cubic [7] adjusts how the sending rate is increased to make it less aggressive. Cubic is now the most successful variation of TCP and widely adopted as the default TCP setting in Linux systems. Modifications to TCP are not ideal as too many variations co-exist, and it is hard to achieve fairness among different versions and the response to congestion can be sluggish.

Alternatively, Active Queue Management (AQM) indicates congestion in the network by dropping packets early, which can give quick responses to congestion. Random Early Detection (RED) [8] which is a classical technique begins to drop the incoming packets when the queue size reaches a threshold and the drop probability is linearly related to the size of the queue. However, RED parameters are pre-configured and cannot adapt to various network conditions. Delay-based adaptive AQMs have emerged recently and Controlled Queuing Delay (CoDel) [9] is the most widely known one for its simplicity and effectiveness.

Even though a whole variety of AQMs and variations exist, most of these focus on wired networks such as adaptive RED (ARED) [10], flow-queue CoDel (fq-CoDel) [11] etc. Evaluations and testing have also been done in wireless networks such as in [4], [12] and [13] but none of them considers the effect of wireless features. Existing AQMs are primarily based on the status of the queue i.e. queue size or the delay each packet suffers in the queue. They either monitor the length of the queue, such as RED, or the queuing delay of packets in the queue, such as CoDel. Efforts have been made to tune AQMs into different network scenarios and traffic types, such as ARED. However, tuning parameters to suit a wide range of scenarios is a challenge and needs to answer several questions e.g. how to quantify a low/high delay, to what extent is delay related to the performance as perceived by the user and are there other metrics such as packet drops and fairness which must be taken into consideration.

In this paper, a Quality of Experience (QoE) based AQM is proposed. It utilizes queueing theory formulation in order to predict queuing delay and makes dropping decisions considering the QoE level over certain period. It is able to reduce the average queuing delay and maintain

Manuscript received March 6, 2018; revised September 18, 2018.
doi:10.12720/jcm.13.10.588-593

target levels of QoE. The rest of the paper is organized as follows. Sections II introduces background and related technology. Section III gives the details of the proposed algorithm. Section IV gives the simulation results and discussion. Section V gives the conclusions.

II. Background and State-of-the-ART

A. Structure of Cellular Networks

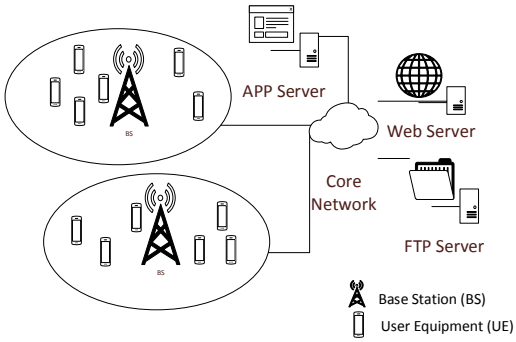


Fig. 1. Cellular network structure

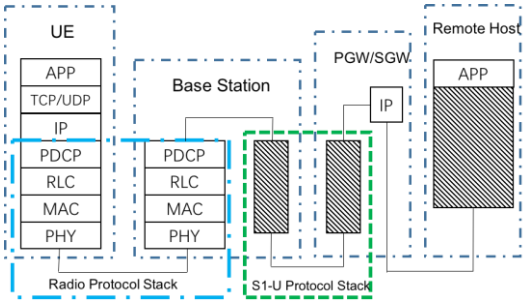


Fig. 2. Protocol stacks of cellular networks

A generic cellular communication system is shown in Fig. 1. Although the link speed of the last hop is increasing with the advance of technology, access networks still contain bottleneck links when there are large numbers of UEs sharing the bandwidth. The protocol stack of cellular networks is different to those of wired and Wi-Fi networks. Data transmitted between the base station and the UE is carried by a virtual concept, "bearer" [14], which means each UE has a dedicated buffer for communication in the base station. As shown in Fig. 2, the Radio Protocol Stack has a Radio Link Control (RLC) layer and Packet Data Convergence Protocol (PDCP) layer, where the queuing of packets for different UEs happens. Each UE, when connected to a base station (BS), will be allocated a dedicated PDCP and RLC buffer for downlink data transmission. The RLC layer has three different transmission modes, Transient Mode (TM), Unacknowledged Mode (UM) and Acknowledged Mode (AM) mode. TM mode and AM mode are often used for signaling and UM mode is mainly used for transmitting data. The proposed algorithm is deployed in the UM buffers so that signaling packets will not be affected.

B. State-of –the-art

An AQM tuned for cellular networks is proposed in [15], and it is a variation of RED and implemented in the RLC layer. The authors change the control function from a linear function to a non-linear one and simulation results show that it outperforms RED from the aspect of average end-to-end delay. However, it is based on RED and cannot solve the tuning issue as the length of the queue is not the sole factor that determines the QoE. [16] and [17] try to control the traffic sending rate by making modifications to the congestion window of the receiver side (rwnd). Both of these take advantage of Round-Trip Time (RTT) and aim to solve Bufferbloat in cellular networks. [16] is primarily based on the estimated RTT and the minimum RTT value. The estimated RTT is the average of RTT value from all the samples of RTT and if the estimated RTT is larger than the minimum RTT, the rwnd will be reduced. [17] controls rwnd by monitoring queue length. The queue length is estimated using the difference between the minimum RTT value and the real RTT value. It assumes that the minimum RTT value is the RTT when there are no packets in the buffer at the access link. However, when there are no packets in the buffer, the dynamic nature of wireless channel and the number of UEs competing for the bandwidth will also affect RTT. Additionally, the calculation of rwnd in [17] is a function of the dropping function of AQM deployed in the router and different AQMs may behave very differently as they have a different dropping policy.

C. G/G/1 Queue and Kingman's Formula

A G/G/1 queue means that the inter-arrival and inter-service time of customers are generally distributed, i.e. they can be any arbitrary distribution [18]. The average queuing delay can be estimated as shown in Eq. (1), where ρ is the load of the queue. C_a and C_s are the coefficient of variation of the inter-arrival and inter-service time distributions respectively. τ is the mean service time per packet.

$$\mathrm{E}(\mathrm{W_q}) \approx \frac{\rho}{1-p}\frac{C_a^2 + C_s^2}{2}\tau \tag{1}$$

D. Quality of Experience (QoE)

As introduced previously, queue length and delay are not ideal performance metrics as they only reflect the networks' congestion level. Cellular networks aim to provide satisfactory services to customers hence how the customers rank the service is directly related to the performance of the network as perceived by the user. From the Internet Services Provider (ISP) perspective, is the proposal of the concept of Quality of Service (QoS) which tries to provide the customer with a required level of service. R factor [19] takes the QoS metrics and returns a QoE measure., as shown in (2) for G.711, where d is the end-to-end delay and e is the loss rate of packets. QoE describes how a user feels about certain type of service under different levels of QoS. According to [20], QoE is more often quantified as the Mean Opinion Source (MOS)

value. The relationship between QoE (as MOS) and R factor is given as in (3).

$$R = 94.2\text{-}0.024d\text{-}0.11(d\text{-}177.3)H(d\text{-}177.3)\text{-}30\ln(1+15e) \quad (2)$$

$$MOS = 1 + 0.035R + 7*10^{6}R(R\text{-}60)(100-R) \quad (3)$$

III. QoE based Queue Management

A. Estimating the Queuing Delay

The proposed algorithm keeps track of the inter-arrival time (λ) and service rate (μ) over a window of 10 packets, which means that it updates the monitored values every 10 packets. The average queuing delay is given by (1). The load and mean service time are given by (4) and (5), where $\bar{\lambda}$ and $\bar{\mu}$ are the average value over the 10 packets window. The square of coefficient of variation of the inter-arrival rate and service rate are given by (6) and (7) respectively.

$$\rho = \frac{\bar{\lambda}}{\bar{\mu}} \quad (4)$$

$$\tau = \frac{1}{\bar{\mu}} \quad (5)$$

$$C_a^2 = Var(\frac{1}{\bar{\lambda}})\bar{\lambda}^2 \quad (6)$$

$$C_s^2 = Var(\frac{1}{\bar{\mu}})\bar{\mu}^2 \quad (7)$$

B. Dropping Policy

The proposed algorithm also tracks the drops at the RLC layer. Estimated queuing delay and packet loss probability are applied into (2) in order to obtain the R factor, while the MOS value is given by (3). According to [19], the MOS factor and the rating of service for Voice over IP (VoIP) traffic is as shown in Table I. According to the required rating level, the decision of whether to drop a packet is made. We assume that when there is no congestion, users will get at least a "high" level of service. When congestion happens, the MOS value starts to degrade due to increasing delay and unexpected drop of packets. When it drops below MOS=4.03, the proposed algorithm checks the MOS value on the arrival of each packets. If the MOS value is still below the upper bound, it will drop 1 packet from the head of the queue. If the MOS value drops below the lower bound, the algorithm stops dropping packets on the realisation that the congestion cannot be solved by actively dropping packets. The degradation of performance may be due to other reasons such as overloading. To guarantee the connection, packets should be kept in the buffer instead of being discarded.

TABLE I: QUALITY RATINGS AND THE ASSOCIATED MOS

Quality of Voice Rating	MOS
Best	4.34-4.5
High	4.03-4.34
Medium	3.60-4.03
Low	3.10-3.60
Poor	2.58-3.10

IV. Simulation Results and Discussion

A. Simulation Setup

The proposed algorithm is implemented in Network Simulator 3 (NS3). The topology used in the simulation is shown in Fig. 3. The UEs are randomly distributed within 500 meters to 5000 meters. The number of UEs varies from 42 to 50 which is a typical value seen in one cell in practice. The buffer at the RLC layer is set to 50 packets. The propagation delay is set to 50 ms and link rate is 100Mbps between the server and the core network. VoIP traffic is generated from the server to UE using the ON-OFF traffic generator, with on time 0.96 seconds and off time 1.69 seconds.

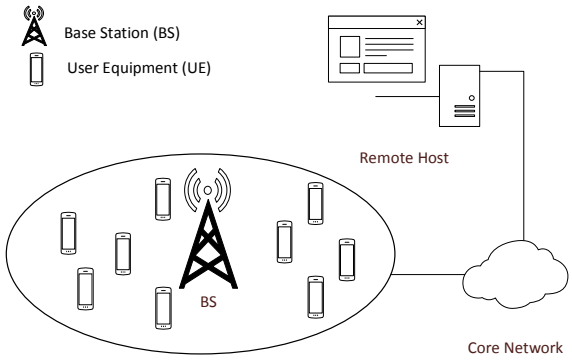


Fig. 3. Simulation topology

B. Implementation of CoDel Algorithm

CoDel is developed for wired networks and it has been proved to work well in Wi-Fi networks [12]. CoDel drops packets from the head of the queue. This needs to be tuned before deployment in cellular networks, as the MAC layer is shared by all the users, and resource are limited. To fully utilize resources, packets can be segmented at RLC layer if the resource from the MAC layer is not enough to transmit a full packet. The remaining segment of the packets will be stored at the head of the queue waiting for the next transmission opportunity.

It is not wise to drop the head packet of the queue in this scenario as it might be the segmentation of a packet; and if it is dropped, there may be no mechanism to recover them. Once TCP is used, the communication will be broken. Hence, the implementation of Codel is slightly altered to check the second packet rather than the first from the head of the queue. Every time a packet is about to be dequeued, it checks the queuing delay of the second packet and if meets the conditions to drop, the second

packet will be dropped hence the head packet of the queue will not be influenced.

C. Simulation Results

The performance of the proposed algorithm is evaluated mainly in terms of the MOS value. As the MOS value is derived from the delay and loss, these two metrics are also evaluated. As shown is Fig. 4, compared with CoDel, the average end-to-end delay is decreased by around 80%. However, this does come at the expense of increasing drops by around 70% as shown in Fig. 5.

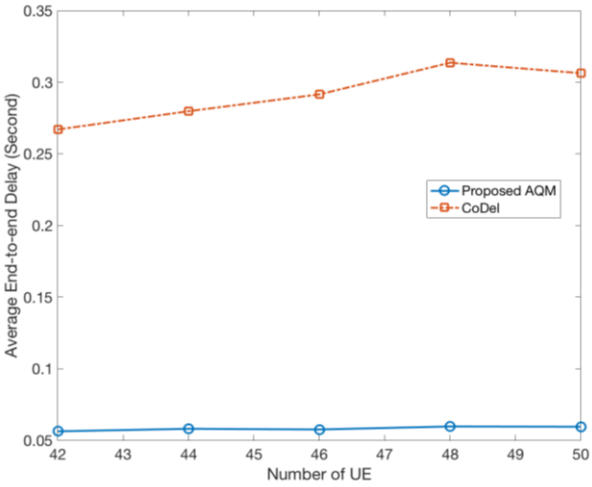


Fig. 4. Average end-to-end delay

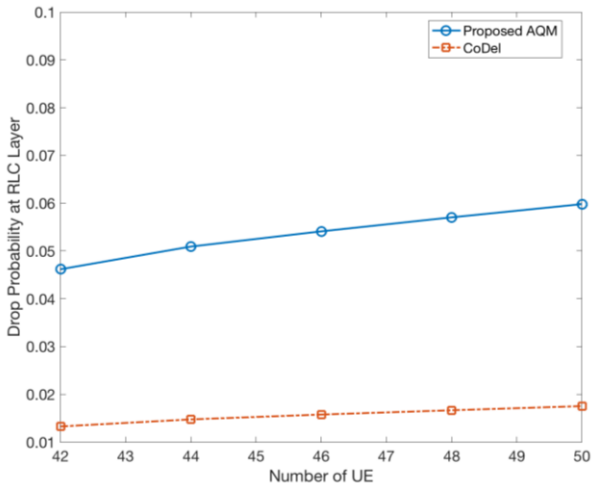


Fig. 5. Drop probability at RLC layer

There is a tradeoff between delay and loss; the MOS factor plays an important role in balancing these two metrics. As shown in Fig. 6 it can be seen that with an increasing number of UEs, the system becomes more congested as delay and loss both increase. And the MOS value decreases for both the proposed algorithm and CoDel. The horizontal line is the lower bound of the medium level service. When there are more than 44 users, CoDel fails to keep the service level. However, the proposed algorithm successfully guarantees the service quality.

Jain's fairness index is used to rate the fairness in a network when there are multiple users [20]. The Jain's fairness index results are shown in Fig. 7 It can be seen that the proposed algorithm maintains similar fairness to that of CoDel.

The strength of the proposed algorithm is summarized below.

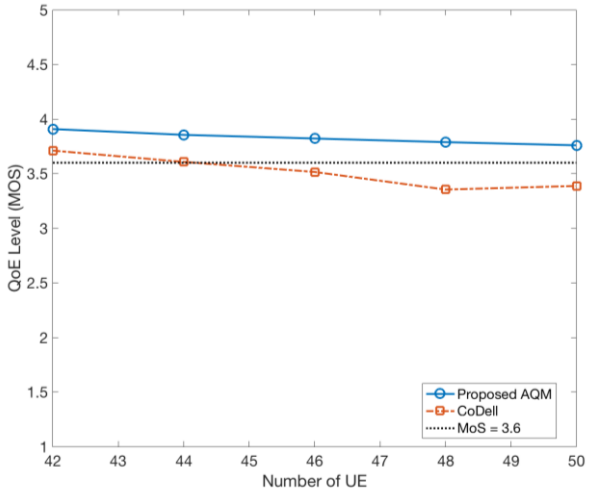


Fig. 6. QoE level

- It monitors the real time inter-arrival and service rate hence it is adaptive and can fit the fast-changing environment in cellular networks.
- The estimation of the queuing delay is solidly based on the classical queuing theory, i.e. G/G/1 queue
- It has only one parameter to be set, i.e. the expected QoE level. With help of QoE metric, delay and loss are automatically balanced. No complicated parameter settings are required.
- The proposed algorithm protects the connection when the system is heavily congested. Real time traffic flows are mostly based on UDP traffic which will not respond to drop of packets. Using other metrics will cause unnecessary drops which doesn't contribute to improving the QoE.

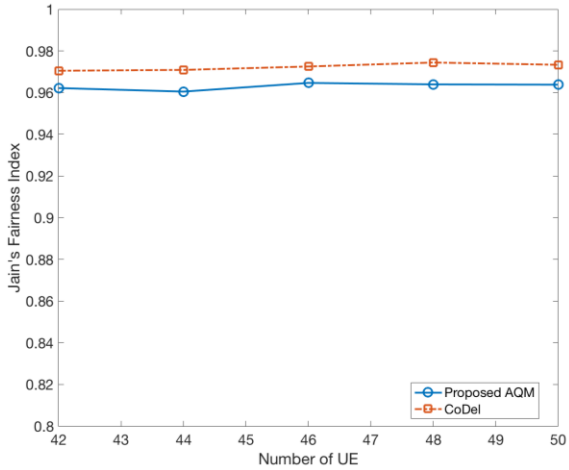


Fig. 7. Jain's fairness index

V. Conclusions

In this paper, a novel user experience based AQM is proposed which aims to improve the QoE of VoIP traffic. Unlike traditional AQMs, the proposed algorithm uses QoE level as a key metric rather than the base metrics delay or queue length. The proposed algorithm is deployed in the base station hence can treat all the UEs equally. Simulation results shows that the proposed algorithm make a good balance between the loss and delay hence improves the QoE for each user.

References

[1] C. V. Forecast, "Cisco visual networking index: Global mobile data traffic forecast update, 2015-2020 white paper," Cisco Public Information, February, vol. 9, 2016.

[2] J. Gettys, "Bufferbloat: Dark buffers in the internet," *IEEE Internet Computing*, vol. 15, no. 3, pp. 96–96, May 2011.

[3] M. Dischinger, A. Haeberlen, K. P. Gummadi, and S. Saroiu, "Char- acterizing residential broadband networks," in *Proc. Internet Measurement Comference*, 2007, pp. 43–56.

[4] T. Høiland-Jørgensen, P. Hurtig, and A. Brunstrom, "The good, the bad and the wifi: Modern {AQMs} in a residential setting," *Computer Networks*, vol. 89, pp. 90–106, 2015.

[5] H. Jiang, Z. Liu, Y. Wang, K. Lee, and I. Rhee, "Understanding bufferbloat in cellular networks," in *Proc. ACM SIGCOMM Workshop on Cellular Networks: Operations, Challenges, and Future Design*, New York, NY, USA: ACM, 2012, pp. 1–6.

[6] L. S. Brakmo and L. L. Peterson, "Tcp vegas: End to end congestion avoidance on a global internet," *IEEE Journal on Selected Areas in Communications*, vol. 13, no. 8, pp. 1465–1480, Oct. 1995.

[7] S. Ha, I. Rhee, and L. Xu, "Cubic: A new tcp-friendly high-speed tcp variant," SIGOPS Oper. Syst. Rev., vol. 42, no. 5, pp. 64–74, Jul. 2008.

[8] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *IEEE/ACM Transactions on Networking*, vol. 1, no. 4, pp. 397–413, Aug. 1993.

[9] K. Nichols and V. Jacobson, "Controlling queue delay," *Queue*, vol. 10, no. 5, pp. 20:20–20:34, May 2012.

[10] A. Showail, K. Jamshaid, and B. Shihada, "Buffer sizing in wireless networks: Challenges, solutions, and opportunities," *IEEE Communications Magazine*, vol. 54, no. 4, pp. 130–137, April 2016.

[11] T. Hoeiland-Joergensen, P. McKenney, D. Taht, J. Ghettys, and E. Dumazet, "Flowqueue-codel: draft-hoeiland-joergensen-aqm-fq-codel-00," Internet-draft, IETF, Tech. Rep., 2014.

[12] T. Jain, A. B., and M. P. Tahiliani, "Performance evaluation of codel for active queue management in wired-cum-wireless networks," in *Proc. Fourth International Conference on Advanced Computing Communication Technologies*, Feb. 2014, pp. 381–385.

[13] A. Showail, K. Jamshaid, and B. Shihada, "An empirical evaluation of bufferbloat in ieee 802.11n wireless networks," in *Proc. IEEE Wireless Communications and Networking Conference*, April 2014, pp. 3088–3093.

[14] F. Capozzi, G. Piro, L. A. Grieco, G. Boggia, and P. Camarda, "Downlink packet scheduling in lte cellular networks: Key design issues and a survey," *IEEE Communications Surveys Tutorials*, vol. 15, no. 2, pp. 678–700, Second 2013.

[15] A. K. Paul, H. Kawakami, A. Tachibana, and T. Hasegawa, "An AQM based congestion control for ENB rlc in 4g/LTE network," in *Proc. IEEE Canadian Conference on Electrical and Computer Engineering*, May 2016, pp. 1–5.

[16] H. Jiang, Y. Wang, K. Lee, and I. Rhee, "Drwa: A receiver-centric solution to bufferbloat in cellular networks," *IEEE Transactions on Mobile Computing*, vol. 15, no. 11, pp. 2719–2734, Nov. 2016.

[17] H. Im, C. Joo, T. Lee, and S. Bahk, "Receiver-side tcp counter measure to bufferbloat in wireless access networks," *IEEE Transactions on Mobile Computing*, vol. 15, no. 8, pp. 2080–2093, Aug. 2016.

[18] P. G. Harrison and N. M. Patel, Performance Modelling of Communication Networks and Computer Architectures (International Computer S, 1st ed. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1992.

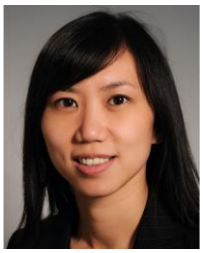
[19] R. G. Cole and J. H. Rosenbluth, "Voice over ip performance monitoring," *SIGCOMM Comput. Commun. Rev.*, vol. 31, no. 2, pp. 9–24, Apr. 2001.

[20] R. Jain, D. M. Chiu, and W. R. Hawe, "A quantitative measure of fairness and discrimination for resource allocation in shared computer system," *Eastern Research Laboratory, Digital Equipment Corporation Hudson*, MA, 1984, vol. 38.

Yuhang Dai received double B.Sc. degrees in Telecommunications Engineering with Management from Beijing University of Posts and Telecommunications, China, and Queen Mary University of London, U.K., in 2014. He is currently pursuing the Ph.D. degree in Electronic Engineering at Queen Mary University of London, London, U.K. His research interests include Bufferbloat and Active Queue Management in Cellular networks.

Vindya Wijeratne, **BENG(Hons)**, **PhD**, **MIET**, **FHEA**. Vindya Wijeratne is a Lecturer in Networks at Queen Mary, University of London, teaching both in London and Beijing, latter under the joint degree programme with Beijing University of Posts and Telecommunications (BUPT). She is also the Director of Teaching for the joint programme. Her main research interests are in enhancing Quality of Experience (QoE) in wireless/mobile networks, scheduling and wireless Software Defined Networks. She has also worked with leading

UK/US/Chinese industry on virtualisation and energy-efficient networks.



Yue Chen (S'02-M'03-SM'15) received her Ph.D. degree from Queen Mary University of London (QM), London, U.K., in 2003. Currently, she is a Professor of Telecommunications Engineering and Computer Science, QM, U.K. Her current research interests include intelligent radio resource management (RRM) for wireless networks; MAC and cooperative wireless networking; HetNets; smart energy systems; and Internet of Things.

John A. Schormans, BSc(ENG), **PhD**, **CEng**, **MIET**, **SFHEA**. John Schormans research interests focus on the application of probabilistic methods to the analysis, simulation and performance measurement of packet-based communications systems and networks. In these subjects, he has published over 140 research papers, supervised to completion 14 PhD students, and has twice been an EPSRC Principal Investigator. John is now working with a London based start-up on user quality of experience in mobile networks via the emulation of key network performance characteristics.