Modeling Guaranteed Delay of Virtualized Wireless Networks Using Network Calculus

Jia Liu[1], Lianming Zhang[1], and Kun Yang[2(✉)]

[1] College of Physics and Information Science, Hunan Normal University, Changsha, China
zlm@hunnu.edu.cn
[2] Network Convergence Laboratory, University of Essex, Colchester, UK
kunyang@essex.ac.uk

Abstract. Wireless network virtualization is an emerging technology that logically divides a wireless network element, such as a base station (BS), into multiple slices with each slice serving as a standalone virtual BS. In such a way, one physical mobile wireless network can be partitioned into multiple virtual networks each operating as an independent wireless network. Wireless virtual networks, as composed of these virtual BSs, need to provide quality of service (QoS) to mobile end user services. One such key QoS parameter is network delay, in particular upper bound delay. This paper presents a delay model for such a wireless virtual network. This delay model considers resources (in particular queues) of both physical nodes and virtual nodes and provides a realistic modelling of the delay behaviours of wireless virtual networks. Network calculus, which usually provides finer insight into a system, is utilized to fulfil the modelling task. The numerical results have shown the effectiveness of the proposed model. The model is useful for both off-line network planning and online network admission control.

Keywords: Wireless network virtualization · Delay modelling · Upper bound delay · Network calculus

1 Introduction

Wireless mobile network virtualization enables physical mobile network operators (PMNO) to partition their network resources into smaller slices and assign each slice to an individual virtual mobile network operator (VMNO) and then manages these virtual networks in a more dynamic and cost-effective fashion [1]. We name these virtualized individual networks as wireless virtual networks or WVNs. VMNOs pay the PMNO in a pay-as-you-use manner. Wireless network virtualization has its real-world bearing in mobile cellular networks. For instance, there are many mobile network operators such as Lebara in the UK that don't have their own physical network infrastructure such as base stations (BS), etc. They typically rent such physical network infrastructure from other MNOs that own the infrastructure. For example, Lebara rents Vodafone's networks to provide mobile services to its own end users. Such mobile users have a Lebara SIM card.

The purpose of virtual networks is to provide services to end users. Services of different traffic characteristics may run on the virtual networks. Therefore some kind of

I. Stojmenovic et al. (Eds.): MOBIQUITOUS 2013, LNICST 131, pp. 602–614, 2014.
DOI: 10.1007/978-3-319-11569-6_47

service differentiation shall be provided in the same way as it is provided in the physical networks. Before carrying out QoS control it is essential to understand behaviours of such virtual networks. Though there is much work on the modelling of physical wireless networks themselves, there is little work on virtual network modelling. Paper [2] presents some initial work on it but the considered network there is mesh networks. Furthermore, there is no closed-formed being deducted in terms of network delay. Nor does this model differentiate physical networks from virtual networks.

As in [1], our paper partitions a physical network node such as a base station into multiple slices. And this partitioning can be carried out in a dynamic manner using software, i.e., supporting software-defined radio networks. Each *slice* represents a virtual network node, e.g., a Lebara BS, as depicted in Fig. 1.

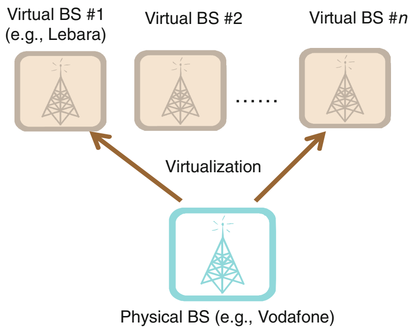

Fig. 1. Wireless network virtualization

As far as network modelling tools are concerned, the predominant ones are probability theory, queue theory. In this paper a new modelling tool called network calculus is utilized. Network calculus is a set of recent developments that enable the effective derivation of deterministic performance bounds in networking [3, 4]. Compared with some traditional statistic theories, network calculus has the merit that provides deep insights into performance analysis of deterministic bounds. Application areas for the network calculus cover a wide range of networking areas such as QoS control, resource allocation and scheduling, and buffer/delay dimensioning [3]. We have carried out some work on using network calculus on wireless sensor networks [5]. This paper extends this previous work into a new research area of network virtualization.

The technical aim of the paper is to propose a delay model for wireless virtual networks (WVNs). In comparison with the existing literature on similar work, this paper has the following major contributions: (1) to systematically analyse the roles in virtual wireless networks and to describe their relationships from a realistic network operation's perspective. The roles include physical networks, virtual networks, end users and service flows. These roles are to be expressed in the network calculus models. (2) To mathematically describe the above virtual wireless network system using network calculus. (3) To propose a QoS model of the virtual wireless network expressing network queue length and network delay in their upper bound terms and in a closed-form manner. The proposed model can help analyse delay guarantee on a per-flow granularity.

In the following sections, we first discuss existing research work on the modelling of virtual networks in Sect. 2. Section 3 presents the system model in terms of different roles including end users, flows, virtual networks (slices) and physical networks, based on which the problem to be addressed in the paper is further clarified. Section 4 models the above NC-formulated system in terms of queue length and delay. After presentation of the numerical results and performance analysis in Sect. 5, the paper is concluded in Sect. 6.

2 Related Work

Network virtualization has emerged as a flexible and efficient way to enable deploying customized services on a shared infrastructure [6, 7]. A lot of research works have developed on wired networks to provide a solution to the gradual ossification problem faced by the existing Internet [8]. Recently, wireless virtualization has attracted increasing attention for its benefits in several interesting deployment scenarios [1, 9, 10], etc. In [9], R. Kokku et al. propose a network virtualization substrate (NVS) for effective virtualization of wireless resources in WiMAX networks. The design provides flow-level virtualization. However, the concept of flows in NVS is not very clear [1]. There is no explanation as to whether a flow refers to an application service flow or packet flow or it represents a type of service.

Though many research works have been carried out on wireless network virtualization, as comprehensively surveyed by [1], there is a lack of the formal modelling of wireless virtual networks. System modelling can provide a useful means to study fundamental features of a system. This paper aims to fill this gap by providing a model of virtualized wireless networks. In particular the focus is given to one important feature of virtual networks, i.e., network delay.

There are various approaches to deal with delay-aware resource control in wireless networks. M. Tao et al. investigate the resource allocation problem in multiuser OFDM system with both delay-constrained and non-delay-constraint traffic [11]. However, the performance impact caused by the delay mechanism is not discussed. There is also an approach which is to convert average delay constraints into equivalent average rate constraints using the queuing theory [12, 13]. However, all these approaches are linked to a particular resource allocation or packet scheduling algorithm and thus are specific to the corresponding algorithms. In our paper, we aim to provide a more generic modelling of wireless virtual networks that is agnostic to resource allocation algorithms and agnostic to a specific network technology. Furthermore, a more expressive means of network modelling tool called network calculus is employed to fulfil the expected network modelling.

Some representative work that uses network calculus to conduct modelling of QoS parameters (in particular delay) is summarized as follows. Paper [14] has utilized network calculus to compute the delay of individual traffic flows in feed-forward networks under arbitrary multiplexing. The maximum end-to-end delay is calculated in [15], again for a feed-forward type of networks. An analytical framework is presented to analyse worst-case performance and to dimension resources of sensor networks [16]. The deterministic performance bound on end-to-end delay for self-similar traffic regulated by a fractal leaky bucket regulator is research into in [17] for ad hoc networks. On the basis of the notion of flows and micro-flows, Zhang et al. [5] propose, using arrival curves and service curves in the network calculus, a two-layer scheduling model for sensor nodes. This piece of work develops a guaranteed QoS model, including the upper bounds on buffer queue length/delay/effective bandwidth. This paper aims to use network calculus to provide a delay mode for wireless virtual networks.

3 System Model Description

3.1 Virtual Network and Virtual Queue

The network illustration of Fig. 1 is depicted in Fig. 2 with the concept of virtual queue. Each slice is allocated a virtual queue in its hosting physical network or network node. All these virtual queues share the data rate capacity of the physical network node, i.e., the physical BS, under the control of a scheduler. The scheduler will take into consideration the QoS requirements of the slices when carrying out resource scheduling. Each slice, as noted as $S1$, $S2$, …, Sn respectively, represents a virtual BS.

The WVN system using virtual queues has the following merits, similar to the virtual buffer concept in [5].

(1) The model provides a minimum guaranteed service rate for every slice under constrained bandwidth. Namely, when a data flow passes through a physical BS it is guaranteed a minimum service rate as specified by its corresponding queue length and the scheduler.

(2) The queues and thus the network bandwidth of a physical BS are shared by all slices. And thus a better gain may be obtained by statistically multiplexing independent slices.

(3) The model simplifies performance analysis, making it suitable for more complex networks such as virtual networks where different roles co-exist.

3.2 Flow Types and Flow Control

Figure 2 illustrates the following two key roles in a network virtualization scenario: physical BS and virtual BS (i.e., slice). Each slice represents a VMN. Each slice has a slice ID. A slice is used by many end users. A user is physically represented by a mobile node in a network. A user may have multiple flows. For example, a user may use his/her smart phone to check emails via listening to an online music piece. Here emails and music pieces each represent a flow. The biggest differentiator amongst flow types is that they have different delay requirement. For instance, voice flow has more stringent delay requirement than non-real time emails. A flow represents a session and each flow has a flow ID. Each uplink packet has both a slice ID showing its belonging to which VMN and a flow ID identifying its service session within a user. The modelling of delay to be carried out in this paper is at a per-flow level.

Figure 3 depicts the relationship of these four key roles in WVNs: physical BS, virtual BS, users and flows. The packets from different users, as denoted as Ui in Fig. 3,

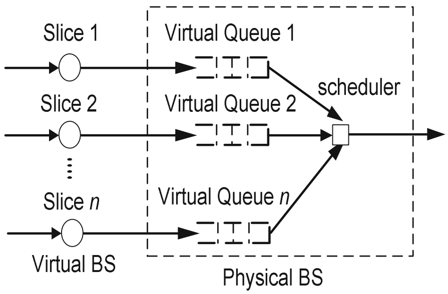

Fig. 2. Virtual network and virtual queue

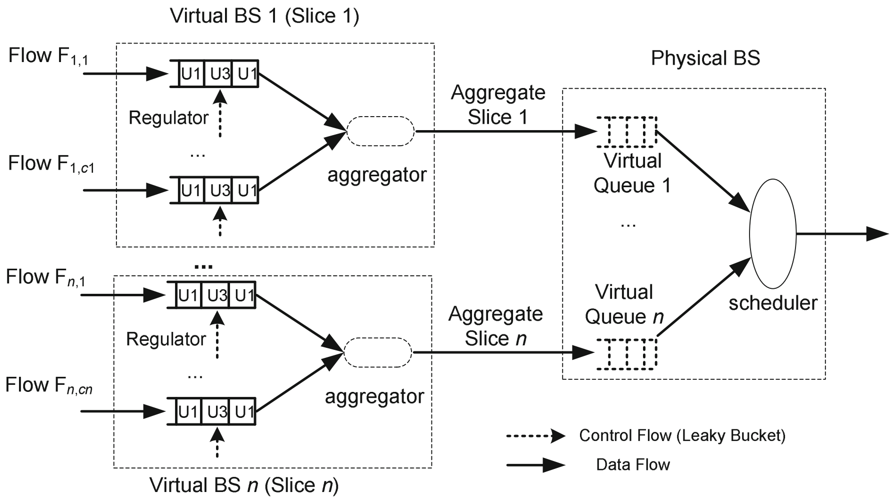

Fig. 3. Flows and slices

but of the same type (e.g., real-time), are put into the same queue in a slice. A leaky bucket source model is selected for each slice queue due to its simplicity and practical applicability. Therefore a leaky bucket regulator is applied to each slice queue to regulate the flows to enable non-rule flows to be controlled under certain conditions. A flow, regulated by the leaky bucket regulator, is indicated by envelope $\alpha(t)$ as shown in Eq. (1) [5],

$$\alpha(t) = r \cdot t + b, \qquad \forall t \geq 0, \tag{1}$$

where b is interpreted as the burst parameter, and r as the average arrival rate.

A flow in an interval $[t, t + \tau]$ is denoted by $A(t, t + \tau)$, and it has the following properties [5].

Property 1 (Additivity): $A(t_1, t_3) = A(t_1, t_2) + A(t_2, t_3), \forall t_3 > t_2 > t_1 > 0$.

Property 2 (Sub-additive Bound): $A(t, t+\tau) \leq \alpha(\tau), \forall t \geq 0, \forall \tau \geq 0$.

Property 3 (Independence): All micro-flows are independent.

4 Proposed Upper Bound Delay Model

In this section, we first describe the above WVN system using network calculus. And then the delay upper bound is deducted based on this WVN model. Before carrying out the above two tasks, some basics regarding network calculus is presented.

4.1 Preliminaries on Network Calculus

Network calculus is the results of the studies on traffic flow problems, min-plus algebra and max-plus algebra applied to qualitative or quantitative analysis for networks in recent years, and it belongs to tropical algebra and topical algebra [5]. Network calculus can be classified into two types: deterministic network calculus and statistical network calculus. The former, using arrival curves and service curves, is mainly used to obtain the exact solution of the bounds on network performance, such as queue length and queue delay, and so on. The latter, on the other hand, based on arrival curves and effective service curves, is used to obtain the stochastic or statistical bounds on the network performance. In this paper deterministic network calculus is utilized as our first step towards modelling WVN as a network calculus system. Some introductory material that is necessary to understand the network calculus modes in this paper is given below. For more information on network calculus please refer to [5, 6].

Theorem 1 (Queue Length and Queue Delay): Assume a slice passes through a physical BS, and the physical BS has an arrival curve $\alpha(t)$ and offers a service curve $\beta(t)$. The queue length Q and the queue delay D of the slice, passing through the physical BS, satisfy the following inequalities, respectively

$$Q \leq \sup_{t \geq 0}\{\alpha(t) - \beta(t)\}, \tag{2}$$

and

$$D \leq \inf_{t \geq 0}\{d \geq 0 : \alpha(t) \leq \beta(t + d)\}. \tag{3}$$

The proof of the theorem and more information about network calculus can be found in [5, 6].

4.2 System Description Using Network Calculus

In this sub-section, we model the WVN presented in Sect. 3 using network calculus. The process of the model is as follows. Firstly, a flow, entering a virtual BS, is regulated by a leaky bucket regulator as given in Eq. (1). Its arrival curve is denoted as $\alpha(t)$. The functions $\alpha(t)$ and $A(t, t + \tau)$ satisfy Property 2. Secondly, we assume the FCFS (first come, first served) strategy is adopted in a queue. This is reasonable as the packets in the same queue are of a same service type. But other more comprehensive queuing strategies may be applied here as well. Finally, the aggregated flows from a slice are scheduled in a way of a service curve $\beta(t)$. The service curve is further explained below.

From Properties 1 and 3, and Fig. 3, the aggregate slices $A_i(t, t + \tau)$ and flows $A_{i,k}(t, t + \tau)$, $k = 1, 2, \ldots, n$ satisfy

$$A_i(t, t+\tau) = \sum\nolimits_{k=1}^{c_i} A_{i,k}(t, t+\tau), \quad \forall t, \tau > 0. \tag{4}$$

From Ref. [18], the equivalent envelope curve $\alpha_i(t)$ of the slices and the envelope curve $\alpha_{i,k}(t)$ ($k = 1, 2, \ldots, n$) of the flows satisfy

$$\alpha_i(t) = \sum\nolimits_{k=1}^{c_i} \alpha_{i,k}(t), \quad \forall t > 0. \tag{5}$$

The service curve $\beta_i(t)$ of Slice i is defined as

$$\beta_i(t) = \beta(t) - \sum_{k=1, k \neq i}^{n} \alpha_k(t - \theta_k), \quad \forall t > \theta \geq 0, \tag{6}$$

where $\beta(t)$ is interpreted as a service curve of physical BS, α_k as an arrival curve of the virtual queue k, and n as the number of the virtual queues(i.e., virtual networks) in the physical BS.

In order to simplify the calculation and without loss of generality, we assume the service curve $\beta(t)$ of the physical BS is a rate-latency function $\beta_{R,T}(t)$ given by

$$\beta(t) = \beta_{R,T}(t) = R \cdot (t - T), \quad \forall t > T > 0, \tag{7}$$

where R is interpreted as the service rate, T as the latency. Obviously for $R > 0$ and $0 \leq t \leq T$, we have $\beta_{R,T}(t) = 0$.

From Property 3, Eq. (5), the simple leaky bucket regulator Eq. (1), the envelope curve of the regulator is

$$\alpha_i(t) = \sum\nolimits_{k=1}^{n} (b_{i,k} + r_{i,k} t). \tag{8}$$

From Eqs. (1) and (7), if $\sum_{i=1}^{n} r_i < R$, the parameter θ_i is optimized, and there is:

$$\theta_i = T + \frac{\sum_{k=1, k \neq i}^{n} b_k}{R}, i = 1, 2, \ldots, n.$$

Bringing θ_i into Eq. (6), and combining Eq. (5) with Eq. (8), there is:

$$\beta_i(t) = \left(R - \sum_{k=1, k \neq i}^{n} \sum_{j=1}^{c_j} r_{k,j}\right) \cdot \left(t - T - \frac{\sum\limits_{k=1, k \neq i}^{n} \sum\limits_{j=1}^{c_j} b_{k,j}}{R}\right). \tag{9}$$

Equation (9) shows that each slice entering the physical BS holds a certain service curve. And the service curve is not only decided by the total service curve of the physical BS scheduler, but also by the arrival curve of the slice.

4.3 The Proposed Delay Model

Now we present, using the network calculus, the guaranteed delay model.

Proposition 1 (Upper Bound on Queue Length): In an interval [0, *t*], the upper bound on Q_i satisfies

$$Q_i = \sup_{t\geq 0}\left\{\sum\nolimits_{k=1}^{n}\left(b_{i,k}+r_{i,k}t\right)-\left(R-\sum_{k=1,k\neq i}^{n}\sum_{j=1}^{c_j}r_{k,j}\right)\cdot\left(t-T-\frac{\sum\limits_{k=1,k\neq i}^{n}\sum\limits_{j=1}^{c_j}b_{k,j}}{R}\right)\right\}. \tag{10}$$

Proof From Eq. (2), we have

$$Q_i \leq \sup_{t\geq 0}\{\alpha_i(t)-\beta_i(t)\}. \tag{11}$$

Substituting Eqs. (8) and (9) into Eq. (11), there is

$$\begin{aligned} Q_i &\leq \sup_{t\geq 0}\{\alpha_i(t)-\beta_i(t)\} \\ &= \sup_{t\geq 0}\left\{\sum\nolimits_{k=1}^{n}\left(b_{i,k}+r_{i,k}t\right)-\left(R-\sum_{k=1,k\neq i}^{n}\sum_{j=1}^{c_j}r_{k,j}\right)\cdot\left(t-T-\frac{\sum\limits_{k=1,k\neq i}^{n}\sum\limits_{j=1}^{c_j}b_{k,j}}{R}\right)\right\}. \end{aligned}$$

Proposition 2 (Upper Bound on Queue Delay): In an interval [0, *t*], the upper bound on D_i satisfies

$$D_i = T+\frac{\sum_{k=1}^{n}b_{i,k}}{R-\sum\limits_{k=1,k\neq i}^{n}\sum\limits_{j=1}^{c_j}r_{k,j}}+\frac{\sum\limits_{k=1,k\neq i}^{n}\sum\limits_{j=1}^{c_j}b_{k,j}}{R}. \tag{12}$$

Proof From Eq. (3), we obtain

$$D_i \leq \inf_{t\geq 0}\{d\geq 0:\alpha_i(t)\leq\beta_i(t+d)\}. \tag{13}$$

Substituting Eqs. (8) and (9) into Eq. (13), there is

$$D_i \leq \inf_{t \geq 0}\left\{ d \geq 0 : \sum_{k=1}^{n} \left(b_{i,k} + r_{i,k}t\right) \leq \left(R - \sum_{k=1,k\neq i}^{n} \sum_{j=1}^{c_j} r_{k,j}\right) \cdot \left(t + d - T - \frac{\sum\limits_{k=1,k\neq i}^{n} \sum\limits_{j=1}^{c_j} b_{k,j}}{R}\right)\right\}$$
$$= \inf_{t \geq 0}\left\{ d \geq 0 : d \geq T + \frac{\sum_{k=1}^{n} \left(b_{i,k} + r_{i,k}t\right)}{R - \sum\limits_{k=1,k\neq i}^{n} \sum\limits_{j=1}^{c_j} r_{k,j}} - t + \frac{\sum\limits_{k=1,k\neq i}^{n} \sum\limits_{j=1}^{c_j} b_{k,j}}{R}\right\}. \tag{14}$$

For $R \geq \sum\limits_{k=1}^{n} \sum\limits_{j=1}^{c_j} r_{k,j}$, from Eq. (14), we obtain

$$D_i = T + \frac{\sum_{k=1}^{n} b_{i,k}}{R - \sum\limits_{k=1,k\neq i}^{n} \sum\limits_{j=1}^{c_j} r_{k,j}} + \frac{\sum\limits_{k=1,k\neq i}^{n} \sum\limits_{j=1}^{c_j} b_{k,j}}{R}.$$

The leaky bucket regulators and aggregators don't increase the upper bounds on queue length/delay of a physical BS, and also don't increase the queue requirements of the physical BS.

5 Numerical Results and Analysis

5.1 Network/Flow Parameter Setup

The two-level model presented in Fig. 3 and Sect. 4 is used for all physical BSs. The service curves $\beta(t)$ of the physical BSs are given in Eq. (9), where R is the service rate and T as the latency of the service curves of the physical BSs. Three slices: Slice1, Slice2 and Slice3, marked as $A_1(t)$, $A_2(t)$ and $A_3(t)$ respectively, are used to showcase the evaluation results. It is assumed that Slice 1 $A_1(t)$ contains three flows: $A_{1,1}(t)$, $A_{1,2}(t)$, $A_{1,3}(t)$, Slice $2A_2(t)$ contains two flows: $A_{2,1}(t)$ and $A_{2,2}(t)$, and Slice $3A_3(t)$ contains one flow: $A_{3,1}(t)$. Here we assume that every flow is regulated by the leaky bucket regulator $\alpha(t)$ as shown in Eq. (8). The average arrival rate $r_{i,k}$ and the burst tolerance $b_{i,k}$ of the six flows are shown in Table 1. The arrival curves of the slices are given by Eq. (9). The unit of queue length Q is Mb. The units of delay D, the time t, and the latency T are ms and the unit of the service rate R is Mb/s except the units that are given.

In terms of evaluation methods, queue length and delay will be investigated against both flow arrival rate and physical BS's service rate. The impact of the traffic burstiness on the network performance is also looked into.

Table 1. Parameters of the three slices and their flows

Slices $A_i(t)$	Flows $A_{i,k}(t)$	Average arrival rate $r_{i,k}$(Kb/s)	Burst tolerance $b_{i,k}$(Kb)
$A_1(t)$	1	240	400
	2	320	360
	3	400	260
$A_2(t)$	1	450	420
	2	350	360
$A_3(t)$	1	700	550

5.2 Queue Length

Figure 4 shows the impact of the average arrival rate r on the queue length upper bounds. Queue length increases as the average arrival rate increases and this is applicable to all slices. Given a fixed service rate, more packets arriving naturally lead to an increased number of packets staying in the queues waiting to be transmitted and thus an increase queue length. As Slice 1 has more traffic to be transmitted – refer to Table 1, its queue length tends to be longer than the other two slices.

Figure 5 depicts the trend of queue lengths of different slices when the service rate R increases. In contrast to Fig. 4, the queue length decreases alongside the increase of service rate and this also applicable to all slices. This is intuitive as serving more packets means a fewer number of packets being buffered in the queues.

Figure 6 shows that the bursty nature of traffic also makes negative impact on network performance. As the level of burstiness increases, as indicated by the burst parameter b, the queue length increases significantly, even more significant than the impact caused by service rate (refer to Fig. 5). It also indicates that the influence of burstiness does not change too much across slices. There is only a small increase in queue length for heavy-load Slice 1 in comparison with Slice 3 which is less loaded.

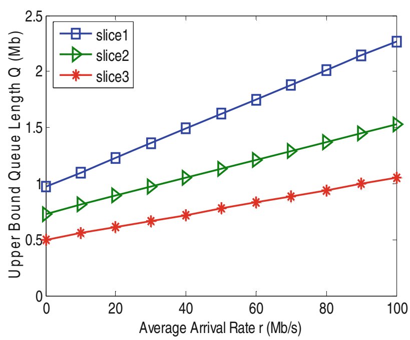

Fig. 4. Upper bound queue length vs. arrival rate (T = 1 ms, R = 100 Mb/s and t = 25 ms)

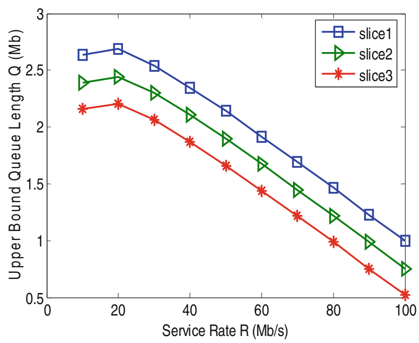

Fig. 5. Upper bound queue length vs. service rate (T = 1 ms and t = 25 ms)

In all these cases other parameters such as service rate R and latency T are fixed as indicated in the corresponding figure captions.

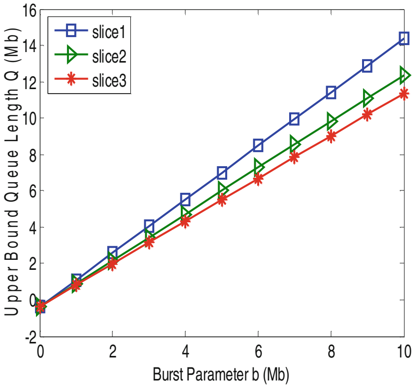

Fig. 6. Upper bound queue length vs. burst parameter (T = 1 ms, R = 100 Mb and t = 5 ms)

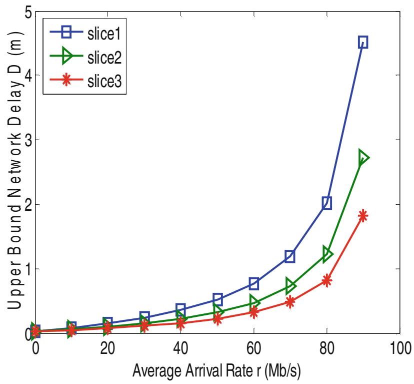

Fig. 7. Upper bound queue delay vs. average arrival rate (T = 1 ms, and R = 100 Mb)

5.3 Network Delay

Figures 7, 8 and 9 show the impact of the same set of variables (i.e., average arrival rate r, service rate R, and burst parameter b) on the upper bounds on queue delay D of a virtual slices. The behavior of network delay more or less follows the similar trend as queue length. It can be observed that the upper bound on D is smaller for larger service rate R.

Figure 7 plots the delay curves as a function of the average arrival rate r. It is obvious that the delay increases as more traffic comes into the system. The delay value starts slow and increases much more significantly as more traffic pours in and thus the network gets more loaded or even congested. This trend applies to all slices though the heavier the slice's load the more obvious the trend is. For instance, Slice 1, which has more load than Slice 2 and Slice 3, suffers more as arrival rate increases.

Figure 8 shows the delay values decrease with the increase of R regardless of T values, converging to almost 0 for all slices. It follows more an exponential trend. For instance, if $T = 1$ ms and $R = 50$, the D value for slices: $A_1(t)$, $A_2(t)$ and $A_3(t)$ is 0.0625, 0.0574 and 0.0452 respectively; and if $T = 1$ ms and $R = 100$ Mb, the D value of the three flows is 0.0315, 0.0290 and 0.0231, respectively. The impact of traffic burstiness on queue delay, as illustrated in Fig. 9, also follows roughly the same trend as its impact on queue length (refer to Fig. 6).

In summary, it can be concluded that the parameters of the flow regulators and the service curves in the physical BS and virtual BS play an important role in modelling a guaranteed delay. In order to achieve network performance improvement and the guaranteed delay for wireless virtual networks, certain mechanisms can be taken to reduce the upper bounds on (end-to-end) delay. These mechanisms may involve designing rational regulator parameters such the average arrival rate and the burst

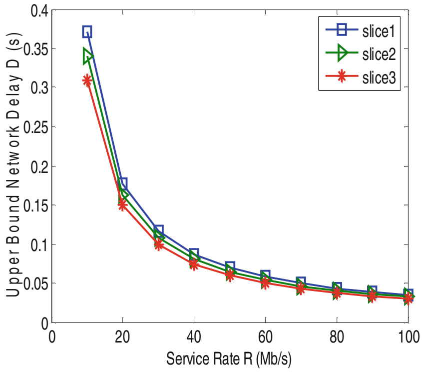

Fig. 8. Upper bound queue delay vs. service rate (T = 1 ms)

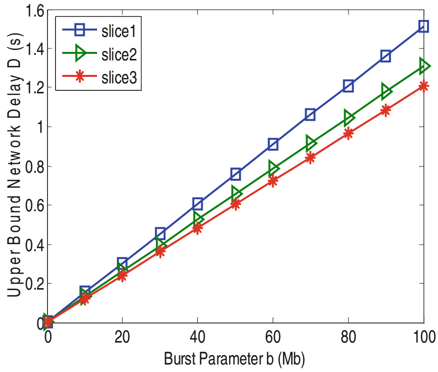

Fig. 9. Upper bound queue delay vs. burst parameter (T = 1 ms, and R = 100 Mb)

tolerance, and designing of rational scheduler parameters such as the service rate and the latency, of the wireless physical networks and wireless virtual networks.

6 Conclusions and Future Work

This paper has proposed a simple but realistic model for describing the upper bound delay of wireless virtual networks. Both service flows that represent service types and virtualized networks as presented by slices are considered in the model. In particular, a finer system modelling and performance analysis tool called network calculus has been adopted to describe the proposed model. Closed-formed formula for the upper bound delay has been deducted, which is useful for either offline wireless network planning or online virtual network admission control. Our immediate future work is to propose a scheduling algorithm based on the above QoS model. Another piece of future work is to extend the model to include other network parameters such as throughput.

References

1. Lu, X., Yang, K., Liu, Y., Zhou, D., Liu, S.: An elastic resource allocation algorithm enabling wireless network virtualization. Wiley Int. J. Wirel. Commun. Mobile Comput. (2012). doi:10.1002/wcm.2342
2. Matos, R., Marques, C., Sargento, S., et al.: Analytical modeling of context-based multi-virtual wireless mesh networks. Elsevier Int. J. Ad hoc Netw. **13**, 191–209 (2011)
3. Le Boudec, J.-Y., Thiran, P.: Network Calculus. Springer, London (2004)
4. Fidler, M.: A survey of deterministic and stochastic service curve models in the network calculus. IEEE Commun. Surv. Tutor. **12**(1), 59–86 (2010)
5. Zhang, L., Yu, J., Deng, X.: Modelling the guaranteed QoS for wireless sensor networks: a network calculus approach. EURASIP J. Wirel. Commun. Netw. **2011**, 82 (2011)

6. Chowdhury, N.M.M.K., Boutaba, R.: Network virtualization: state of the art and research challenges. IEEE Commun. Mag. **47**(7), 20–26 (2009)
7. Schaffrath,G., Werle, C., Papadimitriou,P., Feldmann, A., Bless, R., Greenhalgh, A., Wundsam, A., Kind, M., Maennel, O., Mathy, L.: Network virtualization architecture: proposal and initial prototype. In: Proceedings of ACM VISA, pp. 63–72 (2009)
8. Chowdhury, N.M.M.K., Rahman, M.R., Boutaba, R.: ViNEYard: virtual network embedding algorithms with coordinated node and link mapping. IEEE/ACM Trans. Netw. **20**(1), 206–219 (2012)
9. Kokku, R., Mahindra, R., Zhang, H., Rangarajan, S.: NVS: a substrate for virtualizing wireless resources in cellular networks. IEEE/ACM Trans. Netw. **20**(5), 1333–1346 (2012)
10. Ahn, S., Yoo, C., Network interface virtualization in wireless communication for multi-streaming service. In: International Symposium on Consumer Electronics (ISCE), pp. 67–70, June 2011
11. Tao, M., Liang, Y.C., Zhang, F.: Resource allocation for delay differentiated traffic in multiuser OFDM systems. IEEE Trans. Wirel. Commun. **7**(6), 2190–2201 (2008)
12. Hui, D.S.W., Lau, V.K.N., Wong, H.L.: Cross-layer design for OFDMA wireless systems with heterogeneous delay requirements. IEEE Trans. Wirel. Commun. **6**(8), 2872–2880 (2007)
13. Zarakovitis, C.C., Ni, Q., Skordoulis, D.E., Hadjinicolaou, M.G.: Power-efficient cross-layer design for OFDMA systems with heterogeneous QoS, imperfect CSI, and outage considerations. IEEE Trans. Veh. Technol. **61**(2), 781–798 (2012)
14. Schmitt, J.B., Zdarsky, F.A., Fidler,M.: Delay bounds under arbitrary multiplexing: when network calculus leaves you in the lurch. In: Proceedings of 27th IEEE International Conference on Computer Communications (INFOCOM'08), Phoenix, AZ, USA, April 2008
15. Bouillard, A., Jouhet, L., Thierry, E.: Tight performance bounds in the worst-case analysis of feed-forward networks. In: Proceedings of the 29th IEEE International Conference on Computer Communications (INFOCOM'10), San Diego, CA, USA, pp. 1316–1324, March 2010
16. Schmitt, J.B., Zdarsky, F.A., Thiele, L.: A comprehensive worst-case calculus for wireless sensor networks with in-network processing. In: Proceedings of the 28th IEEE International Real-Time Systems Symposium (RTSS'07), Tucson, AZ, USA, pp. 193–202, December 2007
17. Zhang, L.: Bounds on end-to-end delay jitter with self-similar input traffic in ad hoc wireless network. In: Proceedings of 2008 ISECS International Colloquium on Computing, Communication, Control, and Management (CCCM'08), Guangzhou, China, pp. 538–541, August 2008
18. Fidler, M., Sander, V.: A parameter based admission control for differentiated services networks. Comput. Netw. **44**(4), 463–479 (2004)