Simulation-based analysis of a multi-hop integrated UMTS and WLAN network

Jinglong Zhou · R. Venkatesha Prasad · Yue Lu · Ignas Niemegeers

Abstract The next generation communication networks will comprise of third-generation (3G) and fourth-generation (4G) cellular and multi-hop ad hoc networks. In particular, the Universal Mobile Telecommunications System (UMTS) and the IEEE 802.11 ad hoc networks will play a significant role. The UMTS can provide Internet service in wide areas with excellent mobility support while IEEE 802.11 can provide high speed last hop connectivity in indoor environment. The performance of the offered end-to-end TCP connection when these two different technologies come together to serve the users is an important aspect to be studied. In this paper, we build an analytical model to model the complete process of TCP packet transfer over this integrated network. The delay and end-to-end throughput are modeled. We also verify the simulation results considering the usual error rate in UMTS channels. Our analytical results match well with the simulation models that are widely used. Apart from modeling the Internet connectivity, our model can be used for multitude of tasks, such as gateway selection, resource reservation, etc., in the next generation cellular multi-hop networks.
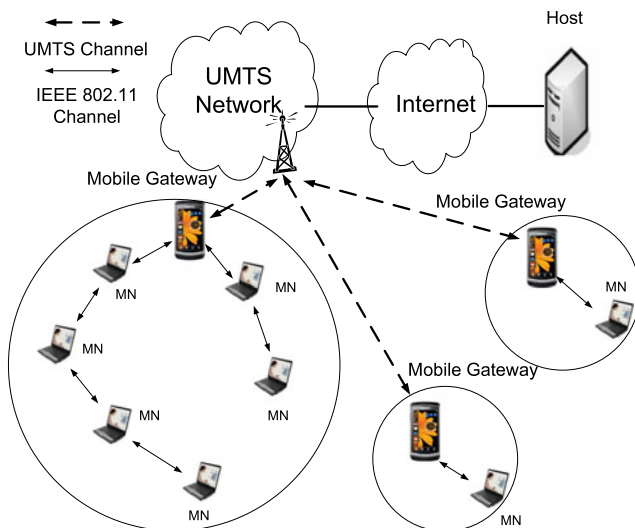
J. Zhou · R. Venkatesha Prasad (✉) · Y. Lu · I. Niemegeers
Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, Mekelweg 4, 2600 GA Delft, The Netherlands
e-mail: R.R.VenkateshaPrasad@tudelft.nl

J. Zhou
e-mail: J.L.Zhou@tudelft.nl

Y. Lu
e-mail: Y.Lu@tudelft.nl

I. Niemegeers
e-mail: I.Niemegeers@tudelft.nl

Keywords UMTS · WLAN · Integrated networks · TCP · Simulation

1 Introduction

Recent fast paced developments in wireless technologies enable mobile electronic devices to connect with each other as well as the Internet using different radio technologies. Already a widespread deployment of the third generation (3G) cellular networks such as Universal Mobile Telecommunications System (UMTS) technology enables high data rate access to the Internet even at a higher speed. Meanwhile, the IEEE 802.11 technology enables short range high speed connection in two modes—infrastructure and ad-hoc modes, usually for indoor users moving at a very low speed.

Both technologies have their own advantages and disadvantages. While 3G cellular networks can provide coverage over a very large area and support high mobility (see Fig. 1), its data rate is still limited. The peak data rate is 14.4 Mbps for its High-Speed Downlink Packet Access (HS-DPA) enhancement. Not every device has a high speed 3G connectivity because of the costs involved. On the other hand IEEE 802.11 can offer large data rate (i.e. 54 Mbps for IEEE 802.11g) and it is popular for its ease of use and low cost amongst many mobile electronic devices such as PDAs. Thus, integrating these two technologies can make these devices which do not have 3G cellular network interface access the Internet.

Let us take a simple yet substantive example. Travelers on a long journey would like to access the Internet for at least at lower speed. Say, in the train there can be few devices with UMTS interface to connect to the UMTS base station. These UMTS devices can act as a gateway with

Fig. 1 An overview of integrated network architecture

both UMTS and 802.11 interfaces. The passengers can use their laptops/PDAs to surf the Internet via their 802.11 interfaces [1]. Thus this integrated network can provide connectivity everywhere and anywhere. Not only the Internet connectivity, we can leverage the advantages of integration of these two technologies and gain many other benefits. Some examples are, load balancing for 3G cellular network, relaying traffic for the device in bad channel condition, and enabling mobile mesh network and mobile hot spots. Further, the nodes in ad-hoc networks can select different relaying nodes and gateways to achieve the best performance on the fly.

Although extensive simulations have been done for some scenarios for single TCP flow in [1], it is hard to use those results in practice because the simulated scenarios are still limited. The simulation work presented can hardly cover all the combinations of topologies and packet error rates. Meanwhile, the packet error rate is also variable in the real environment. The telecom operators can not only rely on the simulation to predict the end-to-end performance for any particular scenario in real time (with many parameters such as number of users, number of hops, different error rates, different TCP configurations).

An analytical model can help us to understand the integrated network in detail. However, for this integrated network, currently, no analytical model is available that can estimate the throughput and decompose the delay for different scenarios. Therefore, the purpose of this work is to provide a faster and simpler method for the telecom operators to calculate the expected end-to-end throughput. Especially, when multiple IEEE 802.11 ad-hoc nodes are added to the integrated network, the model can become complex. The model can be used to investigate the resource reservation, the QoS provisioning and so on. Due to the special bottleneck ef-

fect described in [1], an analytical model can help us to predict the UMTS channel usage and make the proper resource reservation for the UMTS system when a new user in ad-hoc network wants to join the integrated network.

Thus the contributions of this article are, (a) we provide a complete understanding of the TCP over an integrated network explaining various aspects in such a network; (b) a model to present an integrated UMTS and IEEE 802.11 networks; (c) the performance analysis of the integrated network under various conditions are also presented here; and (d) we also provide extensive simulation results. Thus this study will be of immense use for the network planners and service providers for enhancing their services as well as allowing users with only IEEE 802.11 radio to connect to the Internet. We specifically use a simple methodology here. We first decompose the TCP transmission delay into different stages. At each stage, we try to propose an analytical model first and compare it with the thorough simulation studies. Particularly, we use four stages here, RLC bitmap acknowledgement delay, RLC packet delay, TCP roundtrip time and finally the overall throughput. We explain them in sequel. The throughput we are trying to model refers to the saturated throughput with TCP traffic.

This paper is organized as follows. The related work is introduced in Sect. 2. The integrated network architecture and protocols are discussed in Sect. 3. Our analytical model for integrated network and its comparison with the simulation results are discussed in Sect. 4, and the paper concludes in Sect. 5.

2 Related studies

The integrated network architecture described in this paper was first proposed in [1]. Lot of works have also proposed some architectures for integrating the 3G cellular and ad-hoc networks. Hybrid cellular and IEEE 802.11 ad-hoc network architecture is proposed in [2]. Instead of multi-hop, their work focused on one-hop relay in the ad-hoc network. The rationale behind their design was to reduce the system complexity, to avoid the impact of inefficient ad-hoc routing as well as inefficient medium access mechanism. A hybrid cellular and ad-hoc network architecture *viz*, UCAN was proposed in [3] to increase cell throughput without sacrificing the fairness. The proposed architecture requires every mobile station to have both cellular and IEEE 802.11 ad-hoc links. The authors of [4] have described an integrated ad-hoc and cellular network architecture whereby base stations are involved in coordinating peer-to-peer communication in order to increase cell throughput and coverage. However, none of them discusses the end-to-end performance over this new integrated network. A great deal of effort was expended to analyze the performance of transport protocols (Transport Control Protocol (TCP) and User Datagram Protocol

(UDP)) over these two networks separately by means of analytical models, simulations and measurements. Analytical models for the TCP performance over UMTS/HSDPA network and IEEE 802.11 network can be found in [5, 6] and in [5, 7, 8], respectively. The throughput of TCP over UMTS and HSDPA was analyzed in [6]. However only lower downlink speed was modeled. Besides, there is no model for finding the average delay of the Radio Link Control protocol (RLC) packets. The TCP/IP performance over 3G network was analyzed in [9], where the TCP packet delay is analyzed with simulations. A good analytical model was built in [5], which reports the detailed delay composition of the CDMA2000 system. However, the UMTS system described in this paper have a different transmission mechanism. Some other notable studies on modeling and optimization of the RLC layer in the UMTS/HSDPA System are given in [10–13], where the performance of the RLC layer was analyzed using simulations and mathematical models. However, none of them analyzed the integrated network bottleneck effects. To the best of our knowledge, this is the first study which tackles the end-to-end performance of the novel integrated network with an analytical model and compares the results with those of simulation. The throughput modeling for IEEE 802.11 ad-hoc network can be found in [7, 8]. Specifically in our work, we furnish a holistic study of a 3G and IEEE 802.11 network considering end-to-end characteristics. We provide a model as well as show that the model is close to the reality through simulations. In particular, we extend the EURANE model in ns-2 to support UMTS and IEEE 802.11 integrated network. Compared to all the other previous work this study would lead to a methodology to find an analytical model to design an integrated network. It could also be used in optimizing the performance of many protocols and services over such an integrated network.

Recently, more research works have been published focusing the performance of HSPA network [14] and LTE network [15]. It is true that the UMTS technology has been in use for sometime now, and now it is expected that HSPA and LTE will replace it. Meanwhile, the IEEE 802.11 ad hoc network is proposed quite a long ago and not used too much in reality. There are indeed some modeling done indeed for HSDPA and IEEE 802.11 integrated network separately and there are some simulation results in [16]. However, the purpose of this work is to start with some important questions and also using the simulation models correctly. This work is important currently for telecom operators. The machine to machine communication [17], which is growing tremendously currently can benefit from this work. For people's mobile phone, HSPA or LTE network is necessary to provide fast connections, for the machine to machine network, UMTS already can provide a good connection for most of applications. For example, traffic information can be fed to the cars online in the navigation system, since this information may not need high bandwidth channels, UMTS channel

is enough for it. IEEE 802.11 can be used to relay this traffic information for those who does not have UMTS connection, Meanwhile, there are already products like MiFi [18] build the integrated network as we described, on one side it can connect to the UMTS/HSPA network, on another side it can provide WiFi access for up to 5 users, which is sold by the operators to provide data coverage for those devices that provide many services may not have the 3G interface due to the cost or power. However, only one hop 802.11 connection is supported. Then integrated networks like what we have can be greatly useful for those scenarios. Indeed there will be many of these types of heterogeneous networks in future since not all devices would have 3G connectivity. Therefore we feel that there will be many occasions where an integrated network would be useful.

3 Integrated network architecture and simulation setup

In this section, we introduce the architecture of the integrated network and the protocol stacks used in this architecture. Specially, we decompose the TCP round trip time.

3.1 Integrated network architecture and protocols

The integrated network architecture is as shown in Fig. 1. A host is connected to the Internet through IEEE 802.11 radio and the UMTS network. A node that has both UMTS radio interface and IEEE 802.11 radio interface acts as a gateway to connect to the UMTS network and the ad-hoc network. The node in the IEEE 802.11 ad-hoc network can use the gateway to relay the TCP packets and to access the Internet. We can see that both these wireless networks can affect the performance of TCP flows. Based on the bottleneck link, the end-to-end TCP throughput is determined by one of the subnetworks.

To be able to realize the integrated network, we use the following network protocol stacks as depicted in Fig. 2. The IEEE 802.11 ad-hoc network may have different topologies, like star, mesh etc. We use the chain topology to study the impact of the number of hops on end-to-end throughput. We only study the performance up to four hops, since the connection with more than 5 hops will have very small throughput. We can see that the right side of the stack is the UMTS core network, which consists of NodeB, Radio Network Controller (RNC), Serving GPRS Support Node (SGSN) and Gateway GPRS Support Node (GGSN). The UMTS network and the IEEE 802.11 ad-hoc networks are interconnected via a hybrid device called UMTS-802.11 Ad-hoc gateway (referred to as mobile gateway now onwards), which has two different network interfaces as depicted in Fig. 2. To enable the nodes in the ad-hoc network to find and select the mobile gateway, gateway discovery protocol
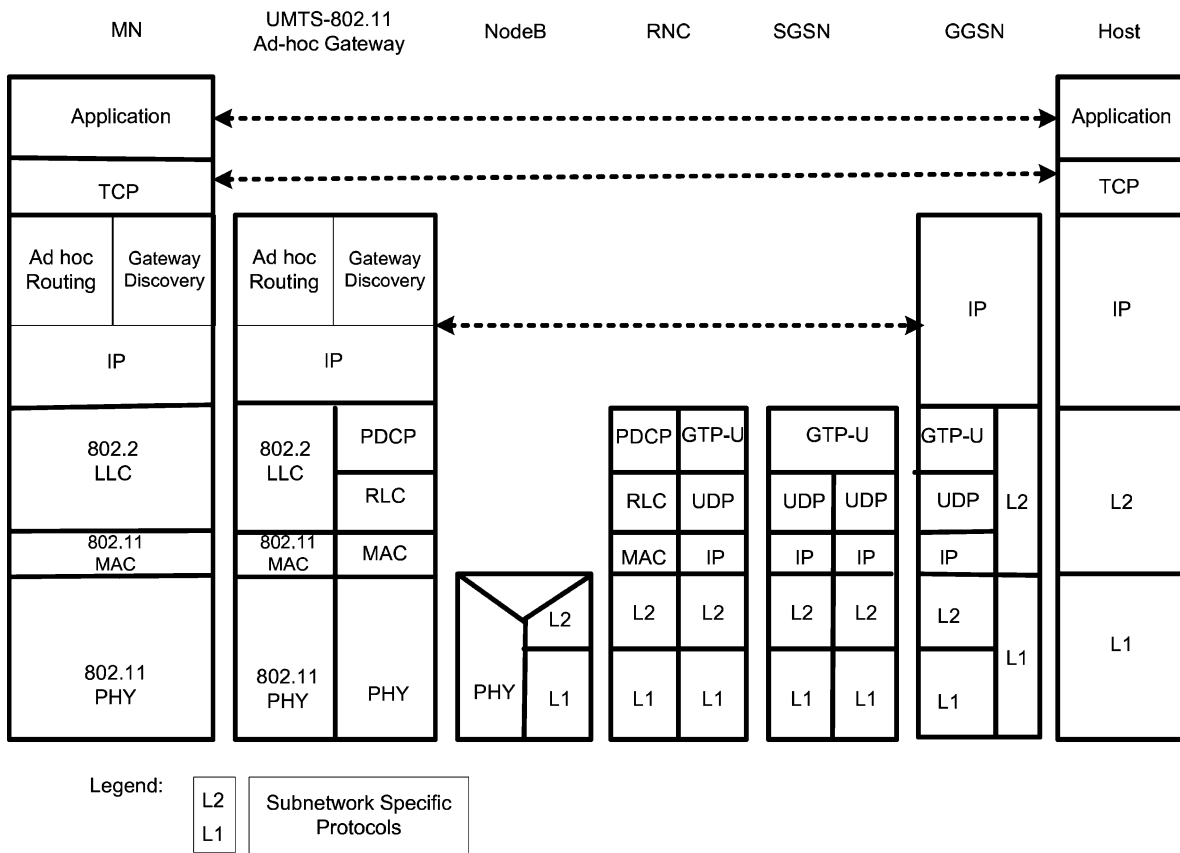
Fig. 2 Protocol stack across the integrated network

is implemented over the IP layer in the mobile gateway. The mobile gateway implements the ad-hoc routing protocol to relay the TCP packets to support the ad-hoc network to connect to the Internet.

3.2 TCP packet transmission procedure

Due to the high loss probability in UMTS links, the lower layer segmentation and retransmission scheme is used. We assume that most of the traffic is on the downlink. It means that the TCP packets are sent from the Internet to the cellular network and then to the IEEE 802.11 network. The TCP packet is segmented into smaller RLC packets, see Fig. 3. The RLC packets can be lost during the transmission over UMTS downlink channel. The receiver side of RLC packet (the gateway) will send acknowledgement consisting of the bitmap of packets that were lost to the RLC sender. This triggers the retransmission process of RLC packets. The RLC packets have to wait for some time before retransmission. Once all the RLC packets within the TCP packets are correctly received, the TCP packet can be further forwarded. In the simulation, we have not set a limit on the number of retransmissions of RLC packets to enable the comparison with the analysis, which is for the sake of simplicity in

analysis.[1] This means, every RLC packet will be transmitted successfully eventually. For the uplink, we assume that there are no errors, which means the TCP acknowledgements will be transmitted only once through the uplink channel. Also, since no packets are really lost, the TCP will work in steady state which allows us to use the bandwidth delay product to model the TCP throughput.

### 3.3 Assumptions for the mathematical model

In this paper, we made some assumptions to make the analysis tractable and simple yet useful to operators. We list them as follows:

3.3.1 Error free IEEE 802.11 channel

In this work, we assume that there is no error in the IEEE 802.11 channel. It is common to see that some links have perfect connection. In [19, Fig. 3], the packet delivery ratio

---

[1]The real UMTS system normally sets the maximum number of retransmission to 10, with the RLC packet error rate used in this paper, the probability that a RLC packet are really lost is quite small and thus can be neglected.

Fig. 3 Decomposition of round trip time of TCP packets
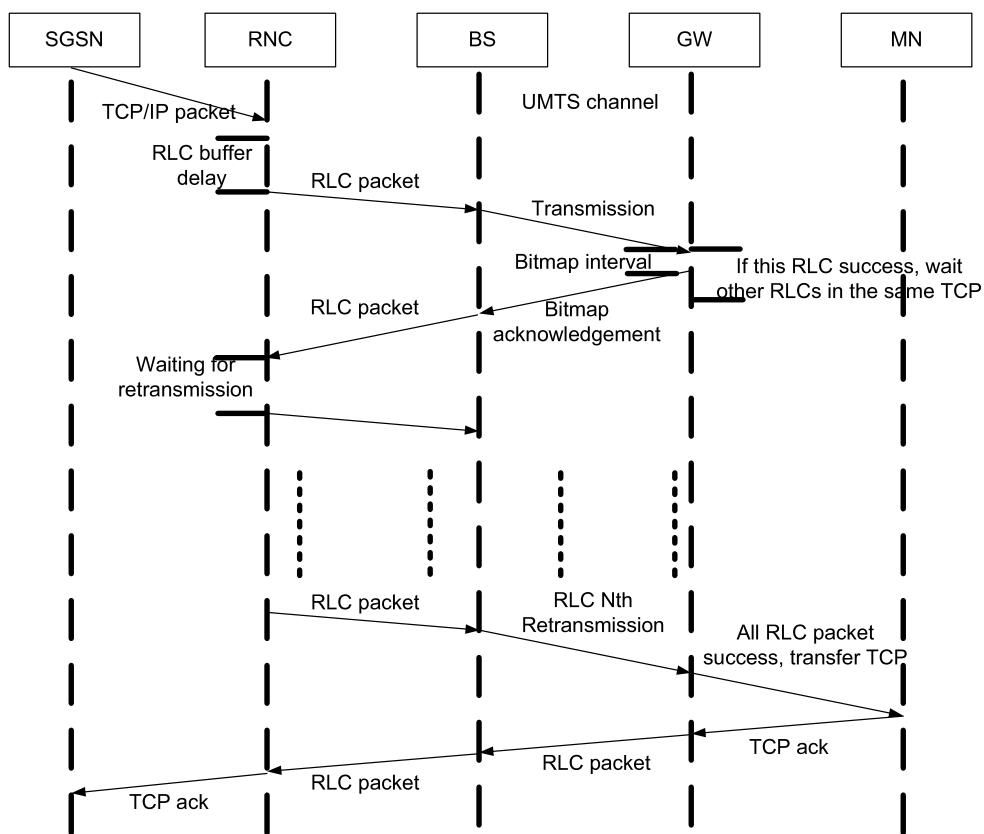
for 2 Mbps can be 1 for certain period despite the severe interfering environment. Meanwhile, if there are packet errors in certain channels, we can also estimate the expected packet delay in the ad hoc network. Even then we can still model the end-to-end throughput.

3.3.2 Error free UMTS uplink

We assume that the UMTS uplink is error free to simplify the model. This is because the network bottleneck is always located at the UMTS downlink. This is because the TCP receiver located in the ad hoc network and the TCP packets are much larger than the acknowledgement packets. We may use coding schemes and other techniques to reduce the packet error rate in the uplink. This is done to simplify the analysis in the paper.

3.3.3 The packet waiting time in RNC buffer time is known from the system

In the model, we get the average waiting time of RLC packets in the RNC from the simulation. This is because, in the real system this information can be easily found.

3.3.4 Only one traffic flow in the system

In this work, we assume only one TCP traffic flow in the network to simplify the model as a beginning. With multiple TCP flows the analysis would be complicated, which is out of the scope of this paper.

3.4 Simulation model

The simulation setup and the topology is as shown in Fig. 4. Our main focus in this article is on UMTS network and since there are many studies which have discussed the modeling of the IEEE 802.11 ad-hoc network in depth, we do not further model the throughput in IEEE 802.11 ad-hoc network explicitly. An extensive research has been reported in [7, 8]. Here we used multi-hop connection in our simulation topology, that means we can choose TCP receiver at 1-hop, 2-hop, 3-hop and 4-hop mobile node. The delay and connection bandwidth for the UMTS network is as shown in Fig. 4. We also list all our simulation parameters in the Table 1.

4 An analytical model of the integrated network

In this section, we present the various stages of our analytical model in the sequel. We also provide a comparison of our

Fig. 4 Simulation topology
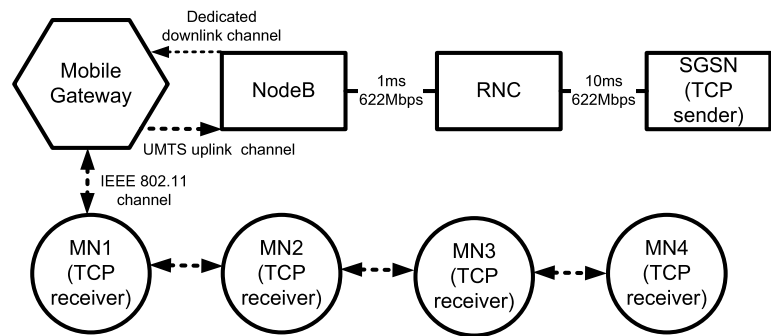
Table 1 Parameters for simulation

	Values
UMTS	
RLC block error rate	0–50
Poll timer	40–100 ms
Status prohibit timer	20–60 ms
Uplink bandwidth	384, 2048 kbps
Downlink TTI	10 ms
Uplink bandwidth	384 kbps
UPlink TTI	10 ms
RLC window size	4096
IEEE 802.11	
IEEE 802.11 bandwidth	11 Mbps
RTS\CTS Threshold	3000 Bytes
TCP Reno	
TCP window size	256 segments
TCP packet size	512 Bytes

analytical model with the simulation results at each stage. While providing the model, we also draw attention to simple yet practical assumptions we have made. To simplify the analysis we first model two networks in this integrated network scenario separately, then we combine them while using the network bottleneck. Since the TCP throughput is indeed sensitive to the bottleneck router, we employ this method here. We assume an error in UMTS RLC layer, but no error in IEEE 802.11 ad-hoc network. In reality there could be some links with perfect delivery, especially when IEEE 802.11 uses low data rate modulations such as 11 Mbps. When there are errors in ad-hoc network, the throughput can still be modeled as in [7, 8]. So we can still combine the two network models in our integrated network to find the end-to-end throughput. The UMTS system uses an ARQ mechanism against the packet loss. However, the end-to-end delay for each TCP packet increases when packets are lost at the RLC, but during the simulation time no timeout event occurs thus only the delay will be increased but it will not

cause TCP timeout. We set the simulation time to be 100 s, and the TCP start up process takes about 3 s. After the start up period, the system comes into the steady state. Since we are interested in the throughput delivered by this integrated network, our model focuses only on the steady state. The definition of all the symbols used in our model can be found in Table 2.

4.1 Throughput modeling for UMTS core network

In this section, based on the decomposition of the TCP packet over integrated network (see Fig. 3), we will model the TCP throughput over UMTS network by four different steps and each step is interdependent. First, we will model the frequency of RLC packet bitmap acknowledgements. Then we model the expected RLC time over UMTS channel. Later we analyze the TCP packet round trip time and finally we use the bandwidth delay product to model the TCP throughput over UMTS network.
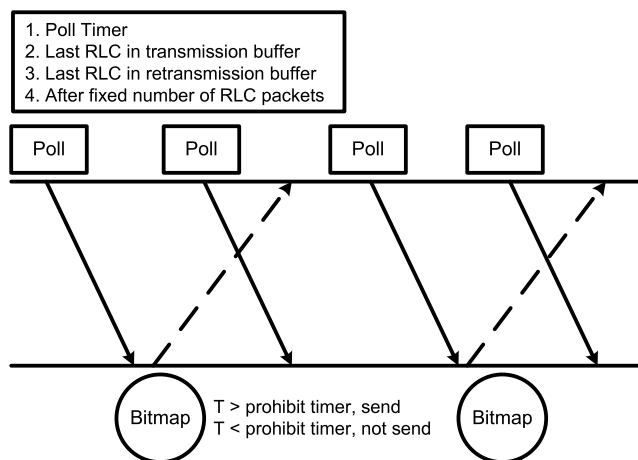
4.1.1 Bitmap acknowledgement

The UMTS system implemented in EURANE Simulator uses the Bitmap acknowledgement method [20]. The transmitter and receiver pair in the bitmap system works in the following way. The transmitter sets a poll bit in a RLC packet based on meeting four different conditions shown in Fig. 5. The status prohibit timer at the receiver side controls when to send a status bitmap message. Once an acknowledged message is sent back to the sender, the prohibit timer will start. If the poll is received during the prohibit time, no status message will be sent out. The status prohibit timer is set to prevent the system to generate redundant retransmissions.[2] However, if the status bitmap is generated too slowly, the channel can not be fully utilized since too many packets are unacknowledged and will be waiting in the transmitter's

---

[2]The redundant retransmission is caused because the bitmap message only indicates a RLC packet at position k is received or not. The RLC packets transmitted in different transmissions are the same; there is no sequencing to differentiate them.

Table 2 Summary of the symbols used

Value	Description	Unit
$P_{RLC}(k)$	the probability that a RLC packet is successfully delivered at kth time transmission	–
p	the RLC packet error rate	–
T_{wait}	the waiting time in RLC buffer	s
T_{bitmap}	the bitmap message coming interval	s
$T_{RLC}(k)$	the total delay for a RLC packet which has k times transmission	s
N	the number of RLC packets able to be transmitted in one TTI	–
$B_{downlink}$	the downlink bandwidth	kbps
q	the percentage of the transmitted RLC packets may be redundantly transmitted	–
L_{buffer}	the length of the buffer in RNC	Bytes
m	the number of RLC packets in a TCP packet	–
$Pr(k)$	the probability that the latest RLC packet arrive at kth transmission	–
$P_{RLC}(n \leq k)$	the probability that a RLC packet can be successfully delivered within k times transmissions	–
$T_{TCP}(k)$	the delay for a TCP packet in which the latest RLC packet arrives at kth transmission	s
V_{RLC}	the RLC packets throughput	Bps
V_{TCP}	the TCP packets throughput	Bps
S_{TCP}	the data size of a TCP packet	Bytes
Sh_{TCP}	the header size of TCP	Bytes
Sh_{ip}	the header size of IP	Bytes

Fig. 5 Bitmap acknowledgement process

buffer. The detailed process of the bitmap acknowledgement is depicted in Fig. 5. In the Fig. 5, T represents the period between the event of receiving current poll message and the time of sending the last acknowledgement.

Rate of the incoming TCP/RLC packet has an impact on the rate of sending polls. The rate of sending polls will impact the bitmap rate in turn and thus affects the TCP transmission rate. The incoming TCP rate is the speed that TCP sender (at the Internet) sends out the TCP packets. The TCP packets may be queued at the RNC node and the length of the queue also keeps on changing based on the difference between TCP and RLC rates. During the whole communi-

cation process, the average value for these two rates are the same, but, the instantaneous value of these two rates might be different. The TCP transmission rate will further impact the TCP process and affects the TCP sending rate from the TCP sender. This is therefore a non-linear problem, which means the delay in delivering TCP packets is affected by the rate at which RLC packet is delivered and vice versa. Therefore, the exact modeling of the process of incoming bitmap is not possible. However, we set quite a low value for the poll timer so that the incoming rate of bitmap acknowledgement at the sender can be approximated to be of uniform distribution. In the simulation, we set different timer values for the status prohibit timer based on different channel bandwidth.[3]

We approximated the bitmap process to a linear process. However, the system behaves differently at $p = 0.0$ and $p > 0.0$. A perfect model could alleviate this problem which is complex. Furthermore, we used the average TCP packet delay from the simulations (as has been done earlier in [22], we use the same method as them to simplify the analysis) which may also induce some approximation due to variance.

4.1.2 RLC packet delay

If the TCP sender is in a fixed network as shown in Fig. 4, for the TCP data packet the RLC transmitter is RNC node itself. Once a TCP packet gets into the RLC transmitter, it

---

[3]Based on [21], the optimal timer value depends on the simulation setting.

is segmented into smaller RLC packets of 40 bytes [20]. These RLC packets are independent of each other. Once all the RLC packets corresponding to a TCP packet arrive at the RLC receiver, the TCP packet is constructed again.

Since we are using the TCP traffic, we always try to analyze the saturation throughput of a TCP flow. The RLC error rate determines the switching of the network bottleneck between UMTS and ad hoc wireless network. Therefore, the RLC error rate is used as x-axis in the following figures. The main focus of this model is to study this so as to provide some tool to the operators to quickly determine the capability of the network to support users.

We assume uniform error distribution and, also we assume the packet error rate to be the same for each transmission. Unlimited number of retransmissions are allowed, which means RLC packet will never be lost but delay will increase if the error rate increases. The rationale behind this assumption is that, in normal channel conditions, the probability that a packet is retransmitted more than 10 times is almost 0. Since channels normally have less than 10% error our model is quite close to the reality. Therefore, assuming unlimited number of retransmission for the sake of making the analysis simple has no bearing compared to the implementation, however there could be a negligible approximation error compared to the analysis. To model the delay distribution of RLC packets, we need to model the probability of number of transmissions for a single RLC packet and the delay for each transmission.

We denote the RLC packet error rate by p, then the probability that a RLC packet will be delivered using only one transmission is $(1 - p)$, and the probability for transmitting twice is $p(1 - p)$. Hence, the probability to transmit the same RLC packet k times before successful delivery is expressed as,

$$P_{RLC}(k) = p^{k-1}(1 - p); \tag{1}$$

We can divide the transmission of an RLC packet into three phases. In Phase-I, the packet is in the transmission buffer. If the first transmission is not successful, the RLC packet will be placed at the end of the retransmission buffer and it will wait for the second transmission, this is the Phase-II. If the RLC packet is transmitted for ≥ 3 times, it will be always in the retransmission buffer and this transmission is said be in Phase-III.

For the Phase-I transmission, the RLC packet will experience the enqueue delay and delay in the transmission stage taking two Transmission Time Intervals (TTI). We use $T_{RLC}(k)$ to denote the delay for an RLC packet which has k transmissions in total. Once the RLC packet is transmitted, the transmitter will move the packet to the retransmission buffer and wait for the bitmap message. After the RLC packets move to the retransmission buffer, the Phase II starts.

If the RLC packet is lost, it will wait in the buffer for retransmission, or else it will be deleted from the buffer. The waiting time in the buffer depends on the buffer length, and the buffer length depends on the channel bandwidth and frequency of receiving the bitmap. We use T_{wait} to denote the waiting time in the RLC buffer. For the Phase-III transmission, since the RLC packet is already in the front of RLC buffer, it only needs to wait for a bitmap message for transmission. The increased delay for the third transmission is equal to T_{bitmap}, where T_{bitmap} represents the arrival interval of bitmap messages. Hence, we can express the total delay for a packet that is transmitted k times, $T_{RLC}(1) = 2 * \text{TTI}$ for $k = 1$; and

$$T_{RLC}(k) = 2 * \text{TTI} * 2 + T_{wait} + (k - 2) * T_{bitmap} \quad \forall k > 1 \tag{2}$$

The average delay of RLC packets can be expressed as,

$$E[T_{RLC}(k)] = \sum_{k=1}^{\infty} P_{RLC}(k) * T_{RLC}(k). \tag{3}$$

We discuss now T_{wait}. The UMTS system offers different downlink channel data rates: 64, 128, 384 and 2048 kbps. We used $B_{downlink}$ to denote the downlink bandwidth. We choose two different typical downlink data rate, 384 kbps and 2048 kbps. The uplink bandwidth is always set as 384 kbps. Now, we have to consider the bitmap interval. As indicated in [13, 21], the timer for bitmap has to be carefully selected in order to gain the highest end-to-end throughput. We set the timer value as shown in Table 1. One bitmap message is 40 Bytes, and each bit can acknowledge one RLC packet. Therefore, one bitmap message can acknowledge a maximum of 320 RLC packets. The number of RLC packets, N, that can be transmitted in one TTI is,

$$N = \frac{\text{TTI} * B_{downlink}}{40 * 8} \tag{4}$$

For each TTI, with 384 kbps channel 12 RLC packets can be transmitted and With 2048 kbps channel 72 RLC packets can be transmitted. Based on the timer setting for 384 kbps, the bitmap can be sent fast enough to acknowledge all the transmitted packets so that there will be no packet accumulating in the RLC transmission buffer. Hence, T_{wait} is determined only by the arrival interval of the bitmap T_{bitmap}.

For the 2048 kbps channel, the bitmap is much more complex. As described in Sect. 4.1.1, if the bitmap comes too fast there will be many redundant retransmissions. Although the end-to-end average delay of RLC may be decreased, the system bandwidth is wasted and the end-to-end TCP throughput is also lower. If the bitmap comes too slowly, the RLC packet end-to-end average delay will increase tremendously. We also validate this by simulation and
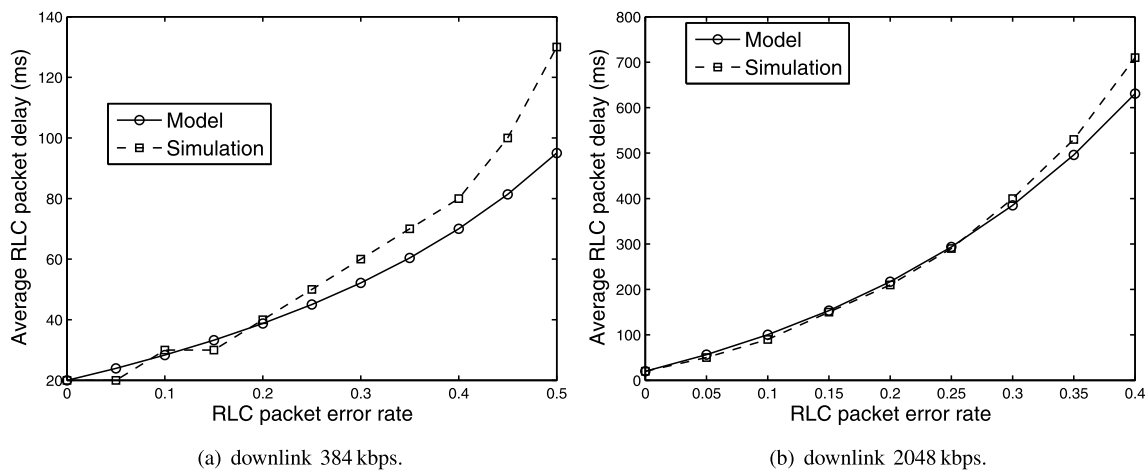
(a) downlink 384 kbps.   (b) downlink 2048 kbps.

Fig. 6 Average RLC packet delay

select the bitmap timer value as in Table 1. The poll timer is set to be larger than the prohibit timer so that it does not push the bitmap too fast, which means that the system will not have redundant retransmissions and the RLC packet arrival and departure rate is the same for the retransmission buffer. This is the steady state.

T_{wait} will be the time that all previous RLC packets have been transmitted. Two factors need to be considered: (1) those retransmitted RLC packets can be lost again during the transmission; (2) q percentage of the transmitted RLC packets may be redundantly transmitted. Hence, T_{wait} can be expressed as,

$$T_{wait} = \frac{L_{buffer} \times (1+p) \times \sum_{k=1}^{\infty} p^k \times 320}{2048 \times 10^3}, \qquad (5)$$

where L_{buffer} represents the buffer length. Based on (3), we use different RLC packet error from 0 to 0.5 to compare the average RLC packet delay found from our model and the simulations (see Fig. 6). For the uplink 384 kbps and the downlink 384 kbps. The RLC delay comparison is as shown in Fig. 6(a) (based on (5)). When the downlink channel rate is 2048 kbps, the result from our analytical model is shown in Fig. 6(b), which matches the simulation results quite well.

4.1.3 TCP roundtrip time

To model the TCP round trip time, we still use the delay distribution of RLC packet delivery. The TCP packet delay in wired network is fixed. In our model, the delay in IEEE 802.11 links is much less than the delay in cellular network or from the Internet, even several retransmissions happen for certain links. Therefore, we do neglect this delay in the total end-to-end delay calculation. The TCP acknowledged packets are sent back by the UMTS uplink channel. Since there is also no error in the uplink link channel, the delay for

acknowledgement is also fixed. There are two parts to the delay, which can be based on various configurations. The buffer delay is at the buffer of RLC node. The TCP transmission delay here also includes the delay caused by RLC segmentation and transmission of RLC packets. This in fact is a non-linear problem. This is due to the fact that there are two loops controlling the behavior of the delay introduced. First, the TCP acting on the measured throughput at its end and controlling the window size and the second one is due to the ARQ mechanism introduced at the RLC. Therefore, we have used the value of the delay measured from the simulation as an input to the model. As described in [22], this will not affect the operational area (region of interest) of the model.

The TCP packet delay in the RLC layer can be modeled as follows. RLC packets are independent of each other and the delay distribution is discussed in Sect. 4.1.2. However, the TCP packet delay is determined by the end-to-end delay of the last RLC packet (of a TCP packet) in the RLC layer. Let us take m RLC packets in a TCP packet. We use $Pr(k)$ to denote the probability that the last RLC packet arrives at kth time of transmission (all RLC packets arrive no latter than k transmissions times).

The probability that for one RLC packet to be delivered successfully is $(1 - p)$, thus for all m RLC packets to be delivered successfully using only one transmission ($k = 1$) is represented by the probability $Pr(1)$ which is equal to $(1 - p)^m$.

For one RLC packet, the probability that it can be delivered within two transmissions is

$$P_{RLC}(n \leq 2) = (1 - p) + p(1 - p) = 1 - p^2. \qquad (6)$$

Continuing, the probability that it can be successfully delivered within k times can be expressed as
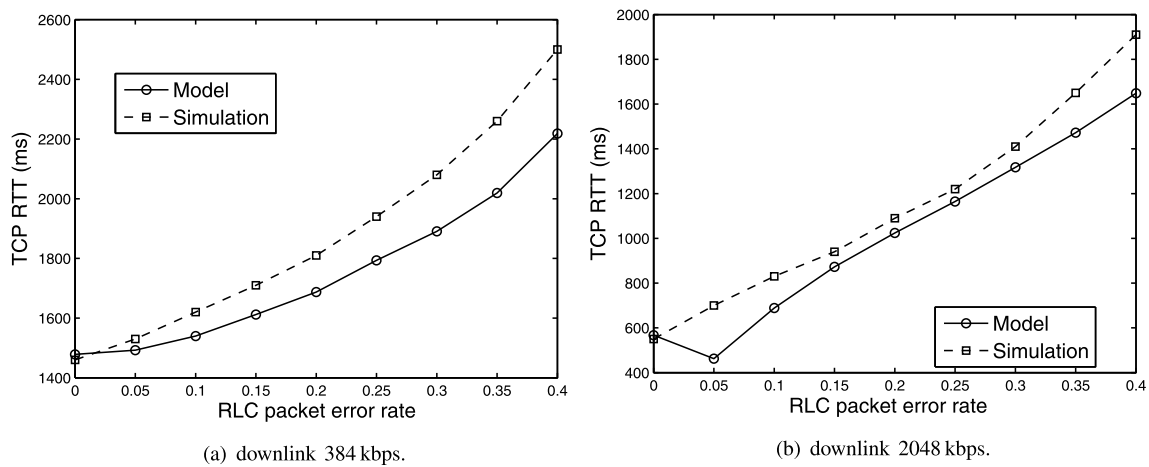
$$P_{RLC}(n \leq k) = 1 - p^k. \qquad (7)$$

(a) downlink 384 kbps.

(b) downlink 2048 kbps.

Fig. 7 Average TCP packet delay comparison with simulations

We can show that

$$Pr(k) = P_{RLC}(n \le k)^m - P_{RLC}(n \le k-1)^m, \qquad (8)$$

where $k \ge 2$. Thus the probability distribution of the number of transmissions of the last RLC packet in a TCP packet is,

$$Pr(k) = (1 - p^k)^m - (1 - p^{k-1})^m \qquad (9)$$

where $k \ge 2$. We can further use the RLC packet delay $T_{RLC}(k)$ to find the average TCP packet delay,

$$E[T_{TCP}(k)] = \sum_{k=1}^{\infty} Pr(k) * T_{RLC}(k). \qquad (10)$$

Based on the (10), we plot the TCP round trip time using the analytical model. The result is compared with the TCP simulation result in Fig. 7 (based on (10)).

4.1.4 Throughput

The TCP bandwidth delay product [23] is a well-known concept in measuring the capacity of a "network pipe", based on the expression *Bandwidth* ∗ *Delay* = *Window* ∗ *Size*, where "*Window*" refers to the TCP window size and "*Size*" refers to the TCP packet size. Since in UMTS network no RLC packet is really lost and the error only adds to the delay of each RLC packet and the delay of the whole TCP transmission in steady state, we can use this relation to estimate the throughput. We use the estimated RTT of a TCP packet since delay and the TCP window size is known in each simulation scenario. We calculated the expected throughput and compared it with the simulation results in Fig. 8 (based on bandwidth delay product). The analytical result matches with the simulation results well. We also calculated the throughput when the downlink channel was 384 kbps and found similar observations.
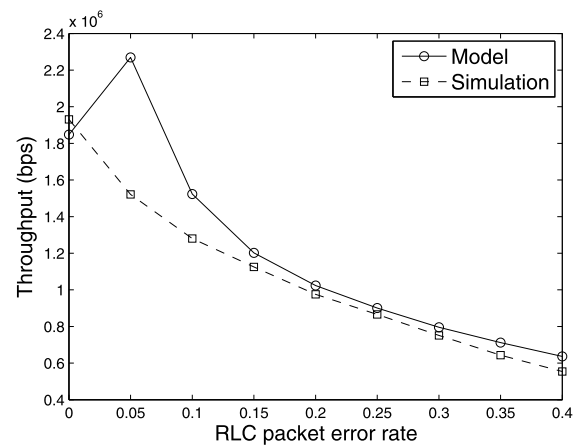

Fig. 8 TCP throughput for downlink channel 2048 kbps

4.2 Modeling throughput for IEEE 802.11 ad-hoc network

The throughput in multi-hop ad-hoc network with IEEE 802.11 radio is explicitly analyzed in many previous studies [7, 8]. We therefore do not expend too many efforts on this part in this article. We explain the rationale behind this assumption as follows. The assumption for single flow is to have a start. Since in both technologies, QoS can be used to differentiate the different flows, we believe if we can use QoS feature, we can still analyze this problem. In any scenario, the network can only have one bottleneck, either in 3G cellular network or ad hoc network. Therefore, the analysis of IEEE 802.11 can be independent of the other network (i.e., UMTS).

One concern may be the collisions from the same network. The simulations of this paper is performed in an ideal environment, which does not have any other IEEE 802.11 nodes in the environment except the experimental nodes. The nodes are specially located at 130 m from each other. The sensing range is set to be 550 m (in reality, this dis-

Table 3 Maximum TCP throughput for IEEE 802.11b with and without RTS/CTS mechanism [1]

Scenario	1-hop	2-hop	3-hop	4-hop
No RTS/CTS	2.26 Mb/s	1.08 Mb/s	0.75 Mb/s	0.56 Mb/s
RTS/CTS	1.31 Mb/s	0.66 Mb/s	0.41 Mb/s	0.32 Mb/s

tance is much less, especially for the indoor scenarios). In the simulation traces (more than 100 s simulation duration), we observe very limit collision events (only very few in the 4-hop simulation traces). As we can see in the simulation results, the network bottleneck is always located at ad hoc network except for one hop scenario. For the 1-hop scenario, there is no collision in the topology. For other scenarios, the throughput we use to predict the integrated network end-to-end throughput is the saturated throughput in the ad hoc network. The simulation results for throughput already contained the collisions event's impact.

Another concern may be the error less channel model in IEEE 802.11. In reality, the wireless channel quality is not the same in all the links and can be quite dynamic for certain channels (see [19]). However, as we stated in Sect. 3.3, we still can estimate the expected packet delay for each hop, for those very dynamic links, we can use the average packet delivery ratio got from longer period of measurement. The end-to-end delay of the integrated network still can be estimated. Finally, we still can model the end-to-end throughput by using those existing models for IEEE 802.11 multi-hop networks which have been proposed and discussed for years.

We list the throughput with some network configurations in Table 3. We combine the two models below.

4.3 Combined model

Based on the bottleneck, the end-to-end throughput is determined by the parts of link which has the lowest bandwidth. The integrated network performance can be found using the combined throughput. Either UMTS or IEEE 802.11 ad-hoc network will be the bottleneck network, while the other sub-network channel is not fully occupied. Therefore, we can use the relation,

$$\mathrm{Tp}_{Total} = Min(\mathrm{Tp}_{UMTS}, \mathrm{Tp}_{802.11}) \qquad (11)$$

to represent the throughput in the integrated network (here we use Tp to represent the Throughput in the equation). The end-to-end TCP throughput is determined by the network which has the lowest throughput. Based on different RLC packet error rates, the UMTS has different throughput and IEEE 802.11 will have constant throughput.[4] For the RLC

---

[4]If there is an error in IEEE 802.11 ad hoc network this conclusion can still be applied, but the IEEE 802.11 model has to be combined together.
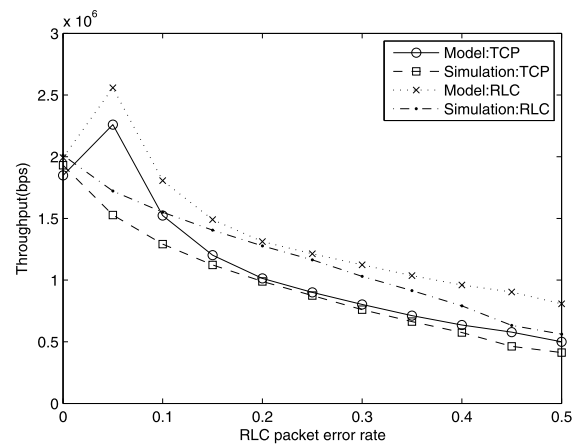
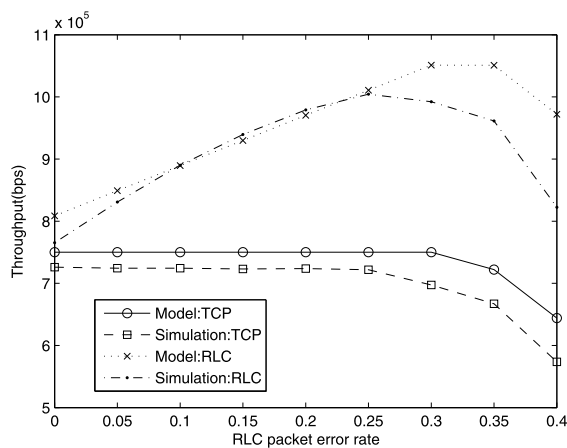**Fig. 9** TCP and RLC throughput for downlink channel 2048 kbps with a 1-hop topology
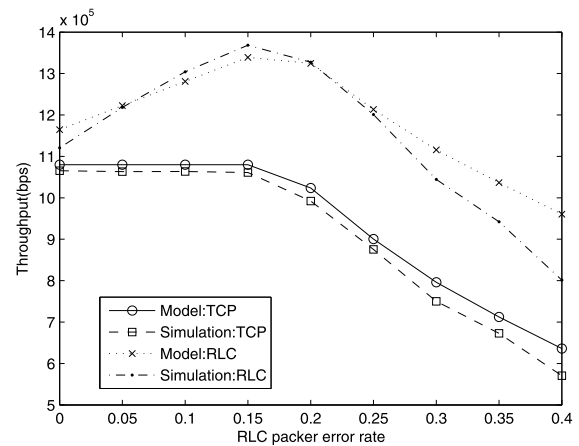
Fig. 10 TCP and RLC throughput for downlink channel 2048 kbps with a 2-hop topology

throughput, we model it in the following way: RLC throughput can be expressed as

$$V_{RLC} = V_{TCP} \times \frac{S_{TCP} + Sh_{TCP} + Sh_{ip}}{S_{TCP}} \times (1 + p), \qquad (12)$$

where V_{TCP} represents the TCP packet throughput, S_{TCP} is the data size of TCP packet, Sh_{TCP} and Sh_{ip} represent the header of TCP and IP individually, and p is the RLC packet error rate. Here, we approximate the redundant RLC retransmissions to be $1 + p$ for small values of p.

We do not assume packet loss in the IEEE 802.11 links, however, if there are packet errors in those links, we can still model them as follows. First, if the UMTS saturation throughput is higher than or equal to the ad hoc network's saturation throughput, the UMTS network is not the bottleneck network and we can calculate the throughput of the ad hoc network and then decide the end-to-end throughput [7, 8]. Secondly, if the UMTS saturation throughput is lower than the saturation throughput of ad hoc network, we can
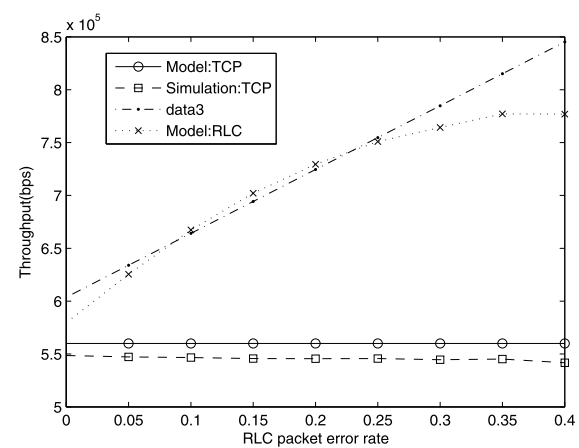
Fig. 11 TCP and RLC throughput for downlink channel 2048 kbps with a 3-hop topology

Fig. 12 TCP and RLC throughput for downlink channel 2048 kbps with a 4-hop topology

add the "Delay" of the TCP packets in the ad hoc network to the total delay. In this paper, we assume that there is no error during the transmission (in some environment, the channel can be very close to this ideal performance), therefore the transmission delay in each hop is much less than the delay in UMTS network. This has also been verified and is indicated in the simulation traces. Therefore, we do not count them in. In the case that there are errors in the IEEE 802.11 channels, we can still use the expected error rate to model the expected delay in the ad hoc network. If we add this delay to the UMTS network, the end-to-end throughput will be accurate too.

We use a multi-hop topology in the integrated network and from the Table 3 we know the throughput for IEEE 802.11 ad-hoc network with number of hops in the ad hoc network. Therefore, we use both analytical model and simulation to get the throughput and show them in Figs. 9, 10, 11, 12 (based on (12)). We can see that our analytical model results for both TCP and RLC throughput match our simulation results very well. It indicates that our analytical model can estimate RLC throughput well.

It is interesting to see that if the bottleneck is at IEEE 802.11, the end-to-end TCP throughput is constant when the UMTS has an error rate between 0 and 0.15. However, as the error rate increases in the UMTS downlink channel, the RLC throughput increases a lot, which means more resources need to be reserved in the channel. Our model can help the system operator to predict the resource consumption when a new user in the IEEE 802.11 network requests a TCP connection. The QoS provision, resource reservation and decision making can be achieved much easily by using our model. The model can facilitate the operator to provide QoS provisioning, resource reservation for single TCP flow or achieve more fairness in resources sharing when multiple TCP flows are in the channel.

In our model, we have used some simplifications. Indeed we have used the standard ns-2 UMTS models especially

validated by the networking community. Since we have two loops that affect the throughput (TCP and the UMTS RLC ACKs), the model is not tractable. Under this circumstance, we have give simple methodology to track the throughput analytically though we have depended on some simplification. The aim of this paper is to show that the simplified analytical expressions could indeed predict the throughput. And this is only verified by simulations as a first step work.

5 Conclusion

In this paper, an analytical model for UMTS and IEEE 802.11 integrated network is proposed to understand the dynamics of TCP packets over two different networks. The TCP packet delay is decomposed into different parts and we compare our analytical model results with the simulation results at each stage. The analytical model matches sufficiently closely with the widely used simulation results. This indicates that our model can be used to predict the TCP throughput over certain network configurations and manage the channel resources well. The results are derived in a manner where the end-to-end paths are decomposed. We have provided extensive simulation results for different data rates too. Moreover, since the results are generic, with slight modifications it can also be used for analyzing other types of traffic such as MPEG-4 streaming over this integrated network. Further, if multiple traffic flows exist in this network, the fairness issue will be an important aspect. Our model can be used for analyzing the fairness amongst different flows and modify the system to reserve and allocate the resources more efficiently. Since newer version of UMTS/HSDPA system is being proposed, we plan to build a model for such versions evolving from the model we have proposed here. We also plan to generalize this framework doing away some of the assumptions made in this article.

References

1. Lo, A., Zhou, J., & Niemegeers, I. (2006). Simulation-based analysis of TCP over beyond 3G cellular multi-hop networks. In *Proc. IEEE PIMRC'06*.
2. Wei, H., & Gitlin, R. (2004). Two-hop-relay architecture for next-generation WWAN/WLAN integration. *IEEE Wireless Communications*, *11*(2), 24–30.
3. Luo, H., Ramjee, R., Sinha, P., Li, L., & Lu, S. (2003). UCAN: a unified cellular and ad-hoc network architecture. In *Proc. ACM Mobicom'03*.
4. Hsieh, H., & Sivakumar, R. (2004). On using peer-to-peer communication in cellular wireless data networks. *IEEE Transactions on Mobile Computing 3*(1), 57–72.
5. Fu, Z., Luo, H., Zerfos, P., Lu, S., Zhang, L., & Gerla, M. (2005). The impact of multihop wireless channel on TCP performance. *IEEE Transactions on Mobile Computing 4*(2), 209–221.
6. Assaad, M., Jouaber, B., & Zeghlache, D. (2004). TCP performance over UMTS-HSDPA system. *Telecommunications Systems*, *27*(2), 371–391.
7. Li, X., Kong, P., & Chua, K. (2005). Analysis of TCP throughput in IEEE 802.11 based multi-hop ad hoc networks. In *Proc. ICCCN'05*.
8. Ng, P., & Liew, S. (2007). Throughput analysis of IEEE802. 11 multi-hop ad hoc networks. *IEEE/ACM Transactions on Networking*, *15*(2), 309–322.
9. Chan, M., & Ramjee, R. (2005). TCP/IP performance over 3G wireless links with rate and delay variation. *Wireless Networks*, *11*(1), 81–97.
10. Harris, J., & Airy, M. (2000). Analytical model for radio link protocol for IS-95 CDMA systems. In *Proc. IEEE VTC 2000-Spring* (Vol. 3).
11. Chatterjee, M., Mandyam, G., & Das, S. (2002). MAC layer retransmissions in 1XTREME. In *Proc. IEEE PIMRC'02*, 15–18 September 2002 (pp. 1452–1456).
12. De Mey, O., Schumacher, L., & Dubois, X. (2005). Optimum number of RLC retransmissions for best TCP performance in UTRAN. In *Proc. IEEE PIMRC'05*.
13. Lefevre, F., & Vivier, G. (2001). Optimizing UMTS link layer parameters for a TCP connection. In *Proc. IEEE VTC'01-Spring* (Vol. 4).
14. Seidel, E. (2006). *Technology of high speed packet access (HSPA)* (Tech. Rep.). Nomor Research GmbH.
15. Motorola (2007). *Long term evolution (LTE): a technical overview* (Tech. Rep.). Motorola, Inc.
16. Zhou, J., Lo, A., Liu, Z., & Niemegeers, I. (2008). TCP performance evaluation over multi-hop cellular network: HSDPA and IEEE 802.11. In *Proc. ISWCS'08*.
17. The FocalPoint Group (2003). *M2M white paper: the growth of device connectivity* (Tech. Rep.). The FocalPoint Group.
18. http://en.wikipedia.org/wiki/MiFi.
19. Zhou, J., Jacobsson, M., Onur, E., & Niemegeers, I. G. (2009). A novel link quality assessment method for mobile multi-rate multi-hop wireless networks. In *Proc. IEEE CCNC'09*.
20. 3GPP Organizational Partners (2000). *3GPP TS 25.322 V3.4.0* (Tech. Rep.)
21. Eurane, http://www.ti-wmc.nl/eurane/.
22. Lin, H., & Das, S. (2005). Performance study of link layer and MAC layer protocols to support TCP in 3G CDMA systems. *IEEE Transactions on Mobile Computing*, *4*(5), 489–501.
23. Stevens, W. (1993). *TCP/IP illustrated (vol. 1): the protocols*. Boston: Addison-Wesley Longman.

Jinglong Zhou received his B.S degrees in Electrical Information Engineering Department from HuaZhong University of Science and Technology, Wuhan, China, in 2004. He received his M.S degree in Delft University of Technology, Delft, The Netherlands, in 2006, graduated with honors. After that, he did four years research in Wireless and Mobile Communication Group (WMC), Delft University of Technology. He got his Ph.D degree in October 2010 from Delft University of Technology. His research interests include Wireless mesh networks, Wireless sensor networks, 3G cellular networks and Personal Networks. He had been evolved in the Dutch Telecommunication project, PNP 2008 project and he won the best student paper award in IEEE CCNC 2009 conference. He is a member of IEEE and he is currently working in Vodafone Netherlands.

R. Venkatesha Prasad received his bachelors degree in Electronics and Communication Engineering and M.Tech degree in Industrial Electronics from University of Mysore, India in 1991 and 1994. He received a Ph.D degree in 2003 from Indian Institute of Science, Bangalore India. During 1996 he was working as a consultant and project associate for ERNET Lab of ECE at Indian Institute of Science. While pursuing the Ph.D degree, from 1999 to 2003 he was also working as a consultant for CEDT, IISc, Bangalore for VoIP application developments as part of Nortel Networks sponsored project. In 2003 he was heading a team of engineers at the Esqube Communication Solutions Pvt. Ltd. Bangalore for the development of various real-time networking applications. Currently, he is a part time consultant to Esqube. From 2005 till date he is a senior researcher at Wireless and Mobile Communications group, Delft University of Technology working on the EU funded projects MAGNET/MAGNET Beyond and PNP-2008 and guiding graduate students. He is an active member of TCCN, IEEE SCC41, and reviewer of many Transactions and Journals. He is on the TPC of many conferences including ICC, GlobeCom, ACM MM, ACM SIGCHI, etc. He is the TPC co-chair of CogNet workshop in 2007, 2008 and 2009 and TPC chair for E2Nets at IEEE ICC-2010. He is also running PerNets workshop from 2006 with IEEE CCNC. He is the Tutorial Co-Chair of CCNC 2009 & 2011 and Demo Chair of IEEE CCNC 2010. He is an invited member of IEEE ComSoc Standards Board and associate editor of European Transaction on Telecommunications.

Yue Lu is currently working in Vodafone Netherlands. She graduated in October, 2010, as a Ph.D. in the Network Architecture and Service group of Delft University of Technology. She completed her MSc Degree from Delft University of Technology in April, 2006, and obtained her BSc Degree at Telecommunication department of HuaZhong University of Science and Technology (China) in 2004. Since 2004, she has been researching P2P systems and real-time multimedia services, via mathematical models and experiments.

Ignas Niemegeers got a degree in Electrical Engineering from the University of Gent, Belgium, in 1970. In 1972 he received a M.Sc.E. degree in Computer Engineering and in 1978 a Ph.D. degree from Purdue University in West Lafayette, Indiana, USA. From 1978 to 1981 he was a designer of packet switching networks at Bell Telephone Mfg. Cy, Antwerp, Belgium. From 1981 to 2002 he was a professor at the Computer Science and the Electrical Engineering Faculties of the University of Twente, Enschede, The Netherlands. From 1995 to 2001 he was Scientific Director of the Centre for Telematics and Information Technology (CTIT) of the University of Twente, a multi-disciplinary research institute on ICT and applications. Since May 2002 he holds the chair Wireless and Mobile Communications at Delft University of Technology, where he is heading the Centre for Wireless and Personal Communication (CWPC) and the Telecommunications Department. He was involved in many European research projects, e.g., the EU projects MAGNET and MAGNET Beyond on personal networks, EUROPCOM on UWB emergency networks and, eSENSE and CRUISE on sensor networks. He is a member of the Expert group of the European technology platform eMobility and IFIP TC-6 on Networking. His present research interests are 4G wireless infrastructures, future home networks, ad-hoc networks, personal networks, cognitive networks.