Harnessing the High Bandwidth of Multi-radio Multi-channel 802.11n Mesh Networks

Ivan Wang-Hei Ho, *Member*, *IEEE*, Patrick P. Lam, Peter Han Joo Chong, *Member*, *IEEE*, and Soung Chang Liew, *Fellow*, *IEEE*

Abstract— There has been an increasing interest in deploying Wireless Mesh Networks (WMNs) for communication and video surveillance purposes thanks to its low cost and ease of deployment. It is well known that a major drawback of WMN is multi-hop bandwidth degradation, which is primarily caused by contention and radio interference. The use of mesh nodes with multiple radios and channels has been regarded as a straightforward solution to the problem in the research community. However, we demonstrate in this paper through real-world experiments that such an approach cannot resolve the multi-hop TCP throughput degradation problem in IEEE 802.11n mesh networks. With extensive experimentation, we verify that the degradation is principally caused by the increase in TCP Round Trip Time (RTT) when the number of hops increases. TCP throughput is fundamentally limited inversely by the RTT. We find that the multi-hop TCP throughput (up to five hops) when using 802.11n is no better than when using 802.11a, despite the much higher data rate 802.11n. We attempt to use multiple parallel TCP connections as a remedy to the problem, and it turns out that the wireless bandwidth can be fully utilized with a sufficient number of parallel streams. In general, our results give a key message that TCP tuning (e.g., setting the correct TCP buffers and use of parallel streams) is of paramount importance in high-bandwidth multi-hop wireless mesh networks that employ the latest wireless standards. These tuning techniques have to be implemented into commercial products to fully leverage the ever advancing wireless technologies to support the growing demand of multi-hop communications in wireless mesh networks.

Index Terms— Wireless Mesh Network; IEEE 802.11n; Multi-Radio Multi-Channel; Multi-hop TCP Throughput; Bandwidth-delay Product.

I. INTRODUCTION

Wireless Mesh Network (WMN) has been attracting a lot of attention from the research community and industry due to its low cost and readily deployable nature. There are also a wide range of deployments and applications of WMNs in the real world [1, 2] for video surveillance, voice communications and localization services.

However, the multi-hop nature of WMN is vulnerable to bandwidth degradation [3 – 5]. This is primarily due to contention or half-duplex communication in single-radio networks. Since a node with a single wireless interface cannot transmit and receive packets at the same time, it must first receive and then transmit in order to relay a packet. Assume the same data rate for every link, the channel time required to relay a packet is thus at least twice the amount of time for sending a packet directly from the source to the destination. Moreover, the carrier-sensing mechanism in 802.11 MAC may prevent simultaneous transmissions on the same channel, and radio interference (or collision) may occur when the carrier sensing fails to prevent interfering links from transmitting simultaneously. These will all affect the multi-hop throughput performance of WMNs.

A number of previous works have proposed the use of multiple radios and channels in wireless mesh networks [6 – 8]. With multiple radios (or wireless interfaces), a node can receive and transmit at the same time on non-overlapping orthogonal channels, and full-duplex communication can then be achieved. The theoretical capacity bounds of multi-radio WMNs were investigated in [9, 10]. Furthermore, with subtle channel allocation, orthogonal channels are assigned to mesh nodes with respect to the network topology to prevent radio interference and allow simultaneous transmissions in the network without any collisions. There are various proposals on channel assignment for multi-radio WMNs to maximize the network capacity [11 – 14]. For instance, [11] presents a joint channel assignment and routing scheme for maximizing the capacity subject to fairness constraints, and [12] proposes distributed and centralized load-aware channel assignment algorithms that make dynamic decisions on a per-flow basis. Hence, the Multi-Radio Multi-Channel (MRMC) approach is a well-accepted solution in the research community to the multi-hop throughput degradation problem, and a

- *I. W.-H. Ho is with the Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong. E-mail: ivanwh.ho@polyu.edu.hk.*
- *P. P. Lam is with P2 Mobile Technologies Limited, Hong Kong Science Park, Shatin, Hong Kong. E-mail: patricklam@p2mt.com.*
- *P. H. J. Chong is with the School of Electrical and Electronic Engineering, Nanyang Technological University, 50 Nanyang Avenue, Singapore 639798. E-mail: ehjchong@ntu.edu.sg.*
- *S. C. Liew is with the Department of Information Engineering, The Chinese University of Hong Kong, Shatin, Hong Kong. E-mail: soung@ie.cuhk.edu.hk.*

Digital Object Indentifier 10.1109/TMC.2013.9 1536-1233/13/$31.00 © 2013 IEEE

number of network planning and communication algorithms were proposed based on the multi-radio architecture in the literature. However, our experimental results indicate that it is not the case for TCP connections in 802.11n WMNs. For instance, we found through real-life experiments that the multi-hop TCP throughput degrades for more than 70% after traversing five hops over an 802.11n mesh network even under some nearly perfect conditions.

The bandwidth-delay product [15] is a useful quantity for analyzing network performance. It represents the number of bytes necessary to fill a TCP path. This quantity also implies that the TCP throughput is limited inversely by the Round Trip Time (RTT) or delay of the communication path given the receiver advertised window size, RWIN (which represents the amount of data that the receiver can accept without acknowledging the sender), as follows:

$$\text{TCP Throughput} \leq \frac{\text{RWIN}}{\text{RTT}}. \quad (1)$$

In this paper, we demonstrate through experimental results that using multiple radios and channels solely is not enough to improve the multi-hop TCP throughput in an 802.11n based WMN. We also verify through control experiments that the multi-hop TCP throughput drop in MRMC 802.11n mesh networks is primarily due to the large RTT of multi-hop wireless communication path and the high bandwidth of 802.11n, resulting in a large bandwidth-delay product where protocol tuning or other remedies are required for achieving the peak throughput. This also explains why we found in our experiments that multi-hop (e.g., five-hop) TCP throughput of 802.11a is similar or comparable to that of 802.11n.

To illustrate how (1) limits the multi-hop TCP throughput of 802.11n, let us consider an example. Assume RWIN = 16 KB = 16,384 bytes and packet size of 1,500 bytes. Then, the RWIN = 16,384/1,500 ≈ 11 packets. We found from our experiment that the average measured TCP data rate of 802.11n is about 80 Mbps for a single-hop wireless link. Hence, the time needed to transmit one packet at the first hop is 1,500*8/80 μs = 0.15 ms. It will take 11*0.15 ms = 1.65 ms before the source transmits all the 11 packets in a TCP window. If at this time, the ACK for the first packet still has not returned from the destination that is multiple hops away, then the TCP source will be idling, and the bandwidth at the first hop will be wasted. Hence, the RTT for the communication path must be smaller than 1.65 ms in order to have no bottleneck. According to our experiments, the RTT for a two-hop wireless path is more than 2 ms, and this is the reason why multi-hop TCP throughput for 802.11n degrades starting from the second hop (more about the results in Section III). On the other hand, for 802.11a, the average measured TCP data rate is only about 20 Mbps. Therefore, the critical RTT that limits the throughput is larger (1500*8*11/20 μs = 6.6 ms). For instance, the five-hop RTT for a wireless path is about 5 ms according to our experiments. Thus, the TCP throughput for MRMC 802.11a can be maintained after traversing five hops in the network.

As a quick remedy to the problem, we attempt to examine the aggregated throughput in a WMN with multiple simultaneously transmitting TCP streams. Our experimental results indicate that the use of multiple parallel TCP connections between the transmitter and receiver that are multiple hops away can better utilize the wireless bandwidth and boost the aggregated throughput. Therefore, TCP tuning techniques such as the use of parallel streams and dynamic adjustment of the advertised window based on the measured behavior need to be enabled in commercial wireless networking products (e.g., via firmware upgrade to enable the window scaling option) in order to fulfill the stringent bandwidth requirement for various real-time applications of wireless mesh network nowadays.

The rest of the paper is organized as follows. Section II describes the experimental setup. The experimental results and discussion of the potential causes of multi-hop TCP throughput drop are presented in Section III. In Section IV, we attempt to use parallel TCP streams for the transmission and measure the network aggregated throughput. Section V discusses proposals and research efforts on the TCP protocol that should be implemented into future wireless commercial products. Finally, Section VI concludes the paper.

II. EXPERIMENTAL SETUP

Figure 1. Experimenting with P2MT's MeshRanger2-o mesh routers.

Let us consider two experiments in Sections II and III. One is for measuring the multi-hop TCP throughput for 802.11n and 802.11a networks (up to five hops). The other one is a control experiment that replaces the wireless links gradually with wired links (up to three links wired) to demonstrate and verify that the large round-trip delay for wireless path is the key factor that limits the multi-hop TCP throughput performance.

Six mesh nodes are used in the experiments. The mesh nodes we used are the MeshRanger2-o mesh routers provided by P2 Mobile Technologies Limited [16] as shown in Fig. 1. Each of the MeshRanger2-o has two dual-band Wi-Fi radio interfaces, in which the Atheros AR7161 chipset is used as the interface controller, capable of operating in either the 2.4 GHz or 5 GHz frequency bands.

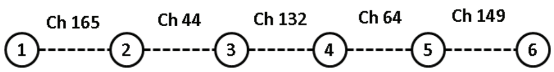

Figure 2. The alignment of the six mesh nodes and the 5 GHz channel assignment.

The six mesh nodes are aligned in a chain topology (with uniform separation of 1.5 m) in an open indoor environment and operating in non-overlapping 5 GHz channels as illustrated in Fig. 2. Note that when the high throughput (HT) operation of 802.11n is enabled, a 40 MHz-wide channel is used. So, the channel assignment needs to take this into account and needs to avoid channel overlapping and radio interference introduced by the "secondary" channel used in the HT operation. All of the nodes use a transmit power of 17 dBm (50 mW) and are equipped with two pairs of omni-directional antennas with a gain of 5 dBi. Note that the results for 802.11n with 20 MHz-wide channel and 802.11n with single antenna are not presented here, since we found that these scenarios also suffer from the same multi-hop throughput degradation problem as 802.11n with MIMO and HT operation, and we have verified that it is not the 802.11n features (such as the 40 MHz-wide channel and MIMO) that undermine the multi-hop throughput performance.

Table 1. Summary of experimental parameters.

Frequency band		5 GHz (Ch 36 – Ch 165)	
# radios per node		2	
Transmit power		17 dBm / 50 mW	
Antenna gain		5 dBi omni-directional	
TCP advertised window size		16 KB	
UDP offered bandwidth		100 Mbps	
Scenarios	11nMRMC	11aMRMC	11nSingle
Channel width	40 MHz	20 MHz	40 MHz
# hops	5	5	5
# channels	5	5	1 (Ch 149)
Normal single-hop TCP throughput	80 – 85 Mbps	20 – 25 Mbps	80 – 85 Mbps

We measure the multi-hop data rate for 802.11a and 802.11n (with 40 MHz-wide channels used in the latter case) with RTS/CTS disabled. Specifically, the network tool, Iperf [17], is used to measure the TCP and UDP throughput for connections with different number of hops, and packets originating from node 1 are transmitted to other nodes in the linear network for the measurements. At the beginning of the experiments, the throughput for every wireless link is examined to ensure that it is working properly. For instance, we found that the normal single-hop TCP throughput for 802.11n is about 80 to 85 Mbps, while that for 802.11a is about 20 to 25 Mbps. In the experiments, the TCP advertised window size is set as 16 KB, while for UDP transmission, the offered UDP bandwidth is 100 Mbps. The ping command is used to get a rough estimation of the RTT for communication paths. According to the experimental results in [18], TCP RTT is slightly smaller than ping RTT for paths destined to light-loaded host. For paths destined to high-loaded host, it depends on an expand coefficient which is the ratio of the mean RTT to the minimum RTT. If the coefficient is small (e.g., less than 20), TCP and Ping RTT are basically the same. Otherwise, TCP RTT appears to be a bit larger. The list of experimental parameters is summarized in Table 1.

III. Experimental Results and Discussion

A. *Multi-hop throughput results and potential factors that affect the multi-hop performance*

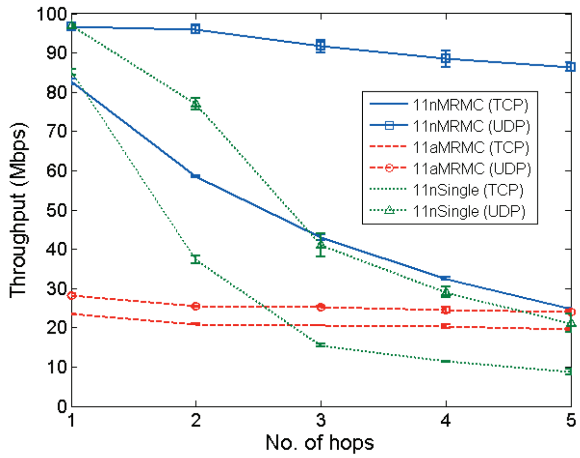

Figure 3. The TCP and UDP multi-hop throughput for 802.11n (MRMC), 802.11n (single channel), and 802.11a (MRMC).

Fig. 3 plots the TCP and UDP multi-hop throughput for single-channel 802.11n (channel 149), Multi-Radio Multi-Channel (MRMC) 802.11n, and MRMC 802.11a. First of all, we can see from the figure that both the TCP and UDP throughput of single-channel 802.11n (green-dotted lines) drop drastically when traversing the network, which is primarily due to radio interference and collision. For its TCP throughput, it is more than halved for every increase of hops. With the use of multiple radios and channels, the TCP and UDP throughput of 802.11a, and the UDP throughput of 802.11n can be sustained over five hops (a fluctuation of less than 10% could be due to the processing of routing overheads or variation of the wireless link quality and is considered to be normal). However, the TCP throughput of MRMC 802.11n (blue-solid line) still drops significantly (over 70%) after five hops. This result is disappointing, especially to researchers that propose the use of multiple radios and channels in wireless mesh networks for boosting the overall network capacity. It is also surprising to note that the TCP throughput of 802.11a (red-dashed line) is almost equal to that of 802.11n after five hops.

In general, there are several potential reasons to account for the multi-hop TCP throughput drop in MRMC 802.11n mesh network that are worth considering.

1. *Radio Interference and Clear Channel Assessment (CCA)*: First of all, if overlapping channels are assigned or channels are reused in the network, co-channel interference and collision might occur. Furthermore, the CCA mechanism in 802.11 consists of Physical Carrier Sensing (PCS) and Energy Detection (ED) [19]. The former detects transmissions of similar systems, while the latter provide information about medium usage. If the receive power level exceeds the corresponding thresholds, PCS or ED will indicate a busy wireless medium and prevent the device from transmitting.

2. *Limited Computation Power of Mesh Points*: The computation power of a mesh router is usually much lower than a personal computer. For instance, the CPU speed of the MeshRanger-2o is 680 MHz and the memory size is 128 MB. The limited computation power of mesh points might incur delay on handling the routing overheads in large-scale wireless mesh network.

3. *Cross-talk and Radio Leakage*: According to [6], due to the near-field effect, inter-radio board cross-talk, radio leakage, and hardware imperfections, self-interference problem may arise when two radios are in close proximity to each other. Since we have two radio cards installed in every mesh point that operate simultaneously, and they are close to each other, inter-radio board cross-talk might happen and degrade the transmission performance.

4. *End-to-end Round-trip Delay*: A large RTT is recorded for multi-hop wireless path in our experiments, which should be incurred by the MAC protocol of 802.11. The large round-trip delay results in a large bandwidth-delay product and restricts the TCP throughput performance as described by (1).

First of all, it is obvious that 1 is not the key factor that limits the multi-hop throughput in our case, since we have already carefully assigned non-overlapping channels for the MRMC mesh network. The results for MRMC 802.11a (both TCP and UDP) and MRMC 802.11n (UDP only) also indicate that co-channel interference and the CCA mechanism are not affecting the data transmission in our set-up at all, otherwise, the multi-hop throughput cannot be sustained and will drop significantly as single-channel 802.11n does. In the following, we are going to validate with a control experiment that Factors 2 and 3 cannot be the primary reason that limits the multi-hop throughput performance either. As a result, 4 becomes the most possible answer for the phenomenon observed in our experiments.

B. *Control Experiment: Results and Discussion*

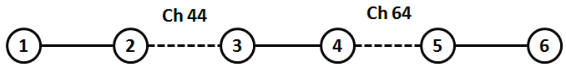

Figure 4. The control experiment with three wired links (at the 1st, 3rd, and 5th hops).

Let us move on to the control experiment. In this experiment, the overall topology and configurations are the same as the previous one, except that wireless links are gradually replaced with wired links. We first wire the link between nodes 1 and 2 (the first hop), then the link between nodes 3 and 4 (the third hop), and finally the link between nodes 5 and 6 (the fifth hop) as illustrated in Fig. 4. After a link is wired, the wireless interfaces that establish the wireless link are disabled to eliminate the possibility of inter-radio board cross-talk within any mesh nodes. The changes in multi-hop throughput and RTT are recorded.

Fig. 5 shows the TCP and UDP throughput across multiple hops when the first hop (or link) is wired (the red dashed lines). We can see that the TCP throughput for the first hop is very high (at 243 Mbps) since nodes 1 and 2 are connected with a wire. It then drops starting from the second hop when traversing through the wireless links. It is interesting to note that the line for the TCP wired case (red-dashed line) appears to lag behind the TCP wireless case (blue-solid line) by one hop. For instance, the two-hop TCP throughput for the wired case is similar to the single-hop TCP throughput for the wireless case (at about 80 – 85 Mbps).

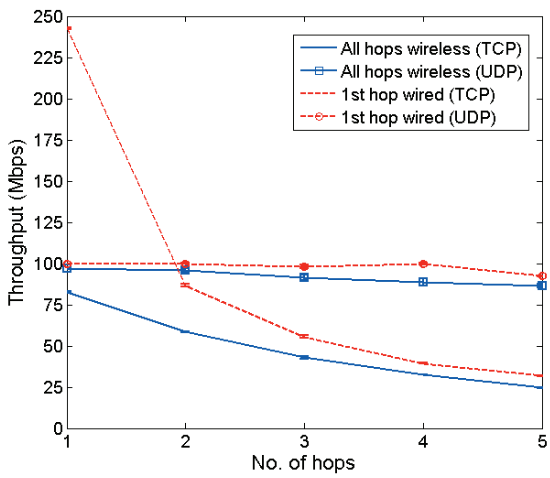

Figure 5. The TCP and UDP multi-hop throughput for 802.11n (MRMC) with all wireless hops and with the first hop wired.

Other than the data rate difference, wired and wireless links also differ in the RTT. Fig. 6 illustrates the differences in multi-hop RTT for the wireless and wired control experiments (with the first, third and fifth hops wired). We can see that the RTT for the wireless case increases almost linearly at a rate of about 1 ms/hop. For the wired case, there is only a slight increase in RTT at the first, third and fifth hops, which indicates that the RTT for wired links is much smaller than that for wireless links. Therefore, according to (1) again, with the RWIN fixed and the RTT increases linearly, the TCP throughput drops at a decreasing rate as shown in Fig. 3.

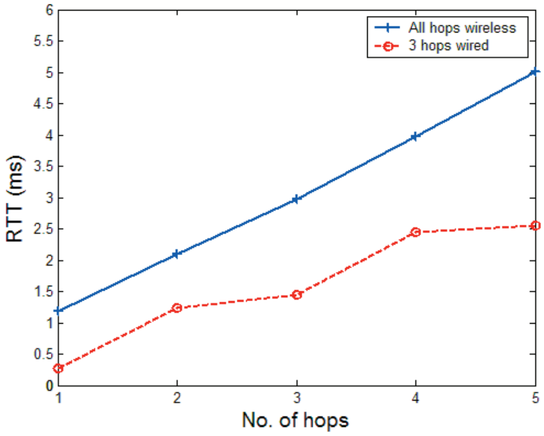

Figure 6. The ping RTT for multi-hop connections with all wireless hops and with three hops wired (at the 1st, 3rd, and 5th hops).

The five-hop TCP and UDP throughput from nodes 1 to 6 for the wireless, 1-hop wired, 2-hop wired, and 3-hop wired scenarios are shown in Fig. 7. There is a small increment in the UDP throughput when the links are wired, it is because wired link is more stable and has less fluctuation than wireless link in general. Note that the UDP throughput is saturated at levels above 90 Mbps, so some minor fluctuation (e.g., between the 2 hops wired and 3 hops wired cases) could be due to some minor variations in the environment between different measurements. For the TCP throughput, it increases significantly with the number of links wired, and is improved by about three times when three hops are wired. According to Fig. 7, it again appears that the "effect" of introducing wired links to the multi-hop network is very similar to reducing the number of hops (or effectively reducing the RTT of the wireless path). For instance, the five-hop path from nodes 1 to 6 in Fig. 4 has similar throughput performance as a two-hop wireless path.

Let us consider the scenario in Fig. 4 with three links wired (the first, third, and fifth hops) and revisit the potential reasons that account for the poor performance proposed in the previous subsection. If the key factor that affects the multi-hop throughput is Factor 2 (limited computation power of mesh points), then the wired control experiment should also suffer from throughput degradation, since both the wired and wireless cases require the processing of routing overheads. But it appears that it is not the case according to the result in Fig. 7. In addition, if Factor 3 (cross-talk and radio leakage) is the key factor, then the five-hop throughput for the three links wired scenario should be able to be maintained at around 80 Mbps (but not 60 Mbps as shown in Fig. 7). Since in that scenario, none of the mesh points has more than one radio card operating simultaneously, hence, the possibility of inter-radio board cross-talk within a mesh node is eliminated.

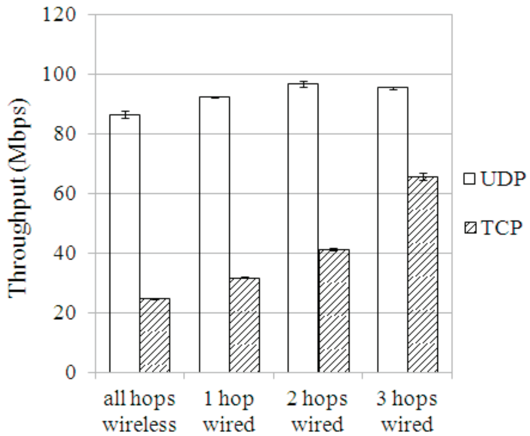

Figure 7. The five-hop TCP and UDP throughput with all wireless hops, one hop wired (the 1st hop), two hops wired (the 1st and 3rd hops), and three hops wired (the 1st, 3rd, and 5th hops).

As a result, it appears that Factor 4 (end-to-end round-trip delay) is the most probable answer to account for the multi-hop TCP throughput degradation phenomenon. The latency (or increase in RTT) should be incurred by the MAC protocol of 802.11. The large RTT for multi-hop wireless path triggers spurious timeouts and retransmissions in TCP and hence leads to poor multi-hop performance.

Table 2 summarizes the multi-hop throughput and RTT results for the MRMC 802.11n network. Let us verify if the results match with the relationship given in (1). Assume that the two-hop TCP throughput is limited by (1), that is, RWIN/RTT = 58.5 Mbps. If this assumption is true, we should be able to predict the throughput for the remaining hops. For example, the estimated throughput for the third hop is 58.5*RTT(2-hop)/RTT(3-hop) = (58.5*2.1)/2.98 = 41.2 Mbps. Our measurement is 43 Mbps, which is pretty close. By the same token, we can estimate the throughput for the fourth and fifth hops, which are (58.5*2.1)/3.97 = 30.9 Mbps and (58.5*2.1)/5.01 = 24.5 Mbps respectively. Again, these predictions match well with our measurements (32.5 and 24.7 Mbps), which further confirms that the multi-hop TCP throughput degradation problem is due to the large RTT of wireless path, and the relationship between the throughput and RTT is described by (1). Furthermore, the phenomenon that the TCP throughput for 802.11a does not drop with the number of hops in our experiment can also be explained with (1). This is because the data rate of 802.11a (about 20 Mbps) is much lower than that of 802.11n (about 80 Mbps with two spatial streams), thus, 802.11a can sustain a much larger RTT until the TCP throughput is limited by (1) (as discussed in the second last paragraph of Section I).

Table 2. The multi-hop TCP and UDP throughput and ping RTT for the MRMC 802.11n mesh network.

# hops	1	2	3	4	5
UDP (Mbps)	97.0	96.8	93.6	91.4	86.9
TCP (Mbps)	82.6	58.5	43.0	32.5	24.7
RTT (ms)	1.19	2.10	2.98	3.97	5.01
Estimated TCP (Mbps)	N/A	N/A	41.2	30.9	24.5

IV. AGGREGATED TCP THROUGHPUT MEASUREMENTS

From the above results, we can conclude that the wireless bandwidth is not fully utilized because of the congestion control mechanism in TCP and the large round-trip delay of the wireless communication path. In this section, we study the aggregated TCP throughput in a wireless mesh network and examine if the approach of using multiple parallel TCP connections can serve as a simple and quick remedy to the multi-hop throughput degradation problem.

A. *Measurement of Aggregated TCP Throughput in a Wireless Mesh Network*

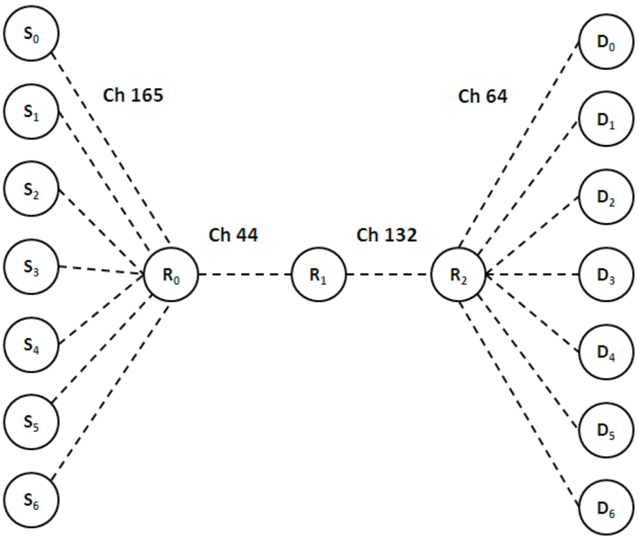

Figure 8. A butterfly-like wireless mesh network.

If the multi-hop bandwidth is limited by the TCP protocol, a wireless mesh network that supports many simultaneous TCP connections (each with small bandwidth requirements) should be achievable. As a whole, these connections may be able to utilize the wireless bandwidth more efficiently. Therefore, it would be interesting to study the aggregated TCP throughput in a wireless mesh network.

Fig. 8 illustrates the butterfly-like 802.11n mesh network for the measurements, in which there are seven source nodes $\{S_0, S_1, \ldots, S_6\}$, seven destination nodes $\{D_0, D_1, \ldots, D_6\}$, and three relay nodes $\{R_0, R_1, R_2\}$. Each node has at most two radios and is assigned with the 5 GHz channels specified in the figure. Packets are transmitted from the source node S_x to the destination node D_x via the three intermediate relay nodes. Hence, there are a maximum of seven four-hop TCP connections transmitting simultaneously in the network. Note that at the first (fourth) hop, the seven source (destination) nodes are connected to the relay node R_0 (R_2) on a single channel, so the relay nodes R_0 and R_2 are bottlenecks in the network, and contentions and packet collisions may happen at these two nodes.

During the experiment, we gradually increase the number of simultaneous transmissions from one to seven and observe the corresponding aggregated TCP throughput, i.e., the total TCP throughput of all simultaneous transmissions in the network. The result is plotted in Fig. 9.

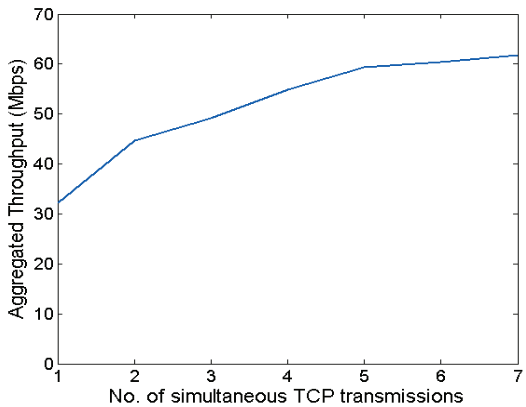

Figure 9. The aggregated throughput in the wireless mesh network with multiple simultaneous TCP transmissions.

We can see from the figure that the four-hop throughput is about 32 Mbps when there is only one TCP transmission, which matches with the (four-hop) result in Fig. 3. As more transmissions participate in the network, the aggregated throughput increases. Note that the aggregated throughput does not increase linearly with the number of simultaneous transmissions, this is because there are collisions and contentions at nodes R_0 and R_2. When there are seven simultaneous transmissions, the aggregated throughput starts saturating at about 62 Mbps (while the TCP throughput for individual connection is 9 Mbps on average), which gives a throughput improvement of more than 90% when compared with the single transmission case. Hence, the results here verify that a wireless mesh network with many simultaneous TCP transmissions can utilize the wireless bandwidth more efficiently than standalone TCP transmission.

B. Potential Solution: Parallel TCP Connections

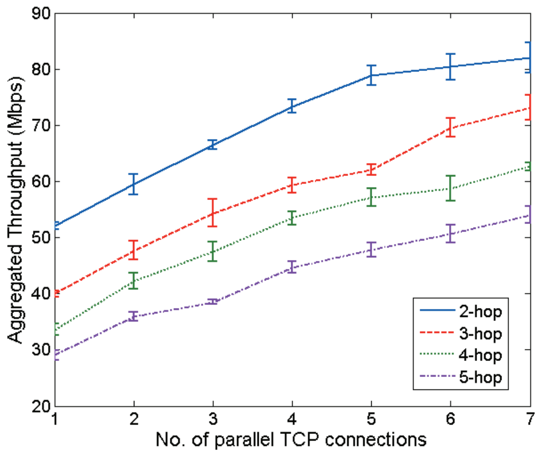

Figure 10. The aggregated throughput for 2-hop to 5-hop paths against the number of parallel TCP connections made.

The result in the previous sub-section suggests that if multi-hop throughput degradation is due to the large bandwidth-delay product of wireless path, then running multiple simultaneous TCP connections could further increase the aggregated throughput. This is because parallel streams take advantage of the fact that TCP tries to share the bandwidth equally among all flows along a path, and n parallel streams will have n times larger aggregated buffer (or advertised window) size. Therefore, parallel connections between any two nodes could be a potential solution for transmitting a large amount of data at high speed from one client to one server through the wireless mesh network.

For the set-up in Fig. 2, Fig. 10 plots the aggregated throughput for two-hop to five-hop communication paths against the number of parallel TCP connections made. We establish parallel connections between the client and server with the Iperf application via the following command on the client side:

```
iperf -c <server> -i 1 -t 60 -P <#connections>.
```

According to the figure, the aggregated throughput grows with the number of parallel TCP connections for all multi-hop paths. With single TCP connection, the two-hop throughput is less than 55 Mbps, which consists with the (two-hop) result in Fig. 3. With seven parallel connections transmitting simultaneously from the client to the server, the aggregated throughput is about 82 Mbps (50% improvement). The result in Fig. 10 again validates that multiple parallel TCP connections can better utilize the available bandwidth of 802.11n and the aggregated throughput is scalable with the number of connections made (up to seven streams).

One way to implement multiple parallel TCP streams is to re-write the application with multiple sockets. Re-sequencing of data segments at the receiver is also needed. There are a number of applications that can use parallel TCP streams. For example, GridFTP [20], bbftp [21], bbcp [22], and Iperf [17]. Alternatively, as the throughput degradation problem is fundamentally caused by the large bandwidth-delay product in (1), we may tackle the problem through rightsizing the TCP advertised window, RWIN. A TCP tuning daemon is proposed in [23] that implements TCP adjustments (including the TCP buffer size and MTU size) based on performance statistics. Since it is a daemon-based mechanism, no modifications to the application are required.

V. RELATED WORKS AND DISCUSSION

TCP was initially designed and optimized for wired networks. Therefore, it is well-known that the fluctuating wireless link quality and user mobility will result in highly variable RTT and delay spikes, and thus incur spurious timeout [24]. There are studies on the design of a more effective 802.11 MAC for improving the throughput of (single-hop) wireless link. For instance, [25] provides a way to distinguish wireless loss versus collision loss at the MAC layer, and assign different back-off mechanisms depending on the type of loss. However, no previous work has addressed the problem incurred by the large RTT in multi-hop wireless networks, primarily because most of the previous studies on wireless TCP and 802.11 MAC are based on single-radio wireless networks (in which bandwidth degradation in multiple hops is expected). Our work here is the first in the literature to point out that the dramatic increase of RTT in multi-hop wireless path could hinder the harness of the high throughput feature (up to 300 Mbps with two spatial streams) in IEEE 802.11n mesh networks, and thus it is important to develop and adopt efficient techniques to further enhance the MAC as well as TCP protocols for multi-hop wireless mesh networks.

There have been a number of methods proposed to improve TCP performance in wireless networks. Generally, they can be classified into two groups. The first group requires new protocol mechanism or modifications to the existing TCP protocol, while the second group of methods do not. TCP Eifel [26] is a typical example in the first group that detects spurious timeouts and retransmissions via implementing time stamping. Another variant of TCP that has been incorporated into Linux is TCP Veno [27], in which the sender side of the TCP connection is modified so as to be less aggressive in reducing the TCP windows during wireless loss. FAST TCP [28, 29] developed by Steven Low's group at CalTech is a TCP congestion avoidance algorithm that targets high-bandwidth high-delay networks (e.g., the MRMC 802.11n mesh network). Modification is required at the sender side for maintaining a constant number of packets in queues via measuring the difference between the observed RTT and RTT when there is no queueing. On the other hand, Split-TCP [30] and related approaches [31 – 34] are solutions in the second group that divide the end-to-end TCP connection into two independent connections with a proxy serving as a common point. In this way, the split TCP connections have smaller RTT, and hence the limitation in (1) can be alleviated. The major drawback of the first group methods is that devices with the newly modified TCP protocols may not be compatible with generic devices. Moreover, some of the proposed algorithms are proprietary and patent protected. For instance, FAST TCP is protected by several patents and being commercialized by the company FastSoft. Although dividing an end-to-end TCP connection into multiple independent segments (such as Split-TCP [30] and Mobile Accelerator [32]) can mitigate the RTT limitation, criticism of Split-TCP and similar methods in the second group is that the TCP ACK received by an application is not actually coming from the end receiver. Thus, it destroys the original semantic that a TCP ACK means the receiver has already received the corresponding TCP packet, and violates the end-to-end security protection. The Snoop Protocol [33, 34] preserves the end-to-end TCP semantics by caching data at the relay AP so as to perform local retransmissions. However, such approach requires modifications of the network-layer software at the relay AP.

Theoretically, a straightforward fix for the issue of large bandwidth-delay product is to increase the advertised window size. In fact, Jacobson et al. proposed the TCP extensions RFC 1323 [35] for high performance, in which new TCP options for scaled windows and timestamps are defined. The TCP window scaling option is an option to increase the TCP advertised window size above its maximum value of 65,535 bytes in order to support communication paths with large bandwidth-delay product. Linux kernels from 2.6.8 have enabled TCP window scaling by default. To make use of this, the send and receive buffer sizes must be increased. Furthermore, we may make use of some daemon-based approaches [23] to dynamically adjust the advertised window based on performance statistics to avoid modifying the applications. TCP window scaling is also implemented in Microsoft Windows since Windows 2000, and is enabled by default in Windows Vista and newer versions. However, if the router or firewall device connected to the Windows machine does not support the TCP extensions, connection errors could occur, and this conflict will cause packet loss and limited connection in many circumstances [36]. On the other hand, the use of parallel TCP streams appears to be a quick remedy to the multi-hop throughput degradation problem according to our experimental results in Section IV, and some existing applications such as GridFTP, bbftp, bbcp and Iperf can support transmissions with parallel TCP streams.

The two proposed solutions have their pros and cons. For instance, the parallel stream solution might be potentially unfair when some applications use more number of streams than others. The applications also need to be revised to re-sequence the data segments, and it increases the load on the end hosts. However, such solution is suitable when the root access to the system for increasing the maximum advertised window size is not available, or some routers or embedded systems in the network are using some old kernel versions that do not support the RFC 1323 extension. Hence, the parallel stream solution appears to be more compatible since we only need to ensure the end hosts are using the same application that supports multiple connections, no matter what operating systems or kernel versions they are running. In general, the system designer and engineer should decide the solution to adopt according to different specifications of networking equipments and systems in the network.

VI. Conclusion

As a conclusion, we have investigated the multi-hop throughput performance of a Multi-Radio Multi-Channel (MRMC) IEEE 802.11n mesh network via experimental studies. Specifically, we have shown that although UDP throughput can be sustained after traversing multiple hops, the TCP throughput degrades dramatically even with the MRMC approach. Our results indicate that the use of multiple radios and channels is not enough to improve multi-hop TCP throughput performance in mesh networks. This is especially prominent in 802.11n with high bandwidth. We have verified that it is primarily because of the congestion control mechanism in TCP and the large round-trip delay of wireless communication path, which makes the bandwidth of the latest WiFi technologies (e.g., 802.11n) not able to be fully harnessed. Actually, there are a lot of research efforts and proposals that address this large bandwidth-delay product issue in high-bandwidth high-delay network. However, it turns out that these solutions have not been incorporated into commercial wireless networking products according to our experimental studies. If wireless product manufacturers do not take actions to implement these solutions, no matter how sophisticated wireless modules are invented and how fast physical data rate can be achieved, the multi-hop performance of wireless mesh networks is still bottlenecked by the protocol.

ACKNOWLEDGEMENT

This work is partially supported by the Hong Kong General Research Funds (Project No. 414812) and the China NSFC grant (Project No. 61271277).

REFERENCES

[1] J. Ishmael, S. Bury, and D. Pezaros, N. Race, "Deploying Rural Community Wireless Mesh Networks," *IEEE Internet Computing Magazine*, vol. 12, no. 4, 2008.
[2] D. Johnson, K. Matthee, D. Sokoya, L. Mboweni, A. Makan, H. Kotze, "Building a Rural Wireless Mesh Network: A do-it-yourself guide to planning and building a Freifunk based mesh network," Meraka Institutue, 2007.
[3] P. C. Ng and S. C. Liew, "Throughput Analysis of IEEE 802.11 Multi-hop Ad-hoc Networks," *IEEE/ACM Trans. on Networking*, vol. 15, no. 2, pp. 309-322, 2007.
[4] C. P. Chan, S. C. Liew, and A. Chan, "Many-to-One Throughput Capacity of IEEE 802.11 Multihop Wireless Networks," *IEEE Trans. on Mobile Computing*, vol. 8, no. 4, pp. 514-527, 2009.
[5] Strix Systems, "Solving the Wireless Mesh Multi-hop Dilemma," White Paper, 2005.
[6] K. Ramachandran, I. Sheriff, E. Belding-Royer, and K. Almeroth, "A Multi-radio 802.11 Mesh Network Architecture," *Mobile Networks and Applications*, vol. 13, no. 1-2, 2008.
[7] Y. Liu, Y. Xiong, Y. Yang, P. Xu, and Q. Zhang, "An Experimental Study on Multi-channel Multi-radio Multi-hop Wireless Networks," IEEE Global Communications Conference (GLOBECOM), 2005.
[8] C. Zhang, K. Kowalik, and M. Davis, "An Experimental Study of the Impact of Using Multi-radio in WLAN Mesh Networks," Wireless Communications, Networking and Mobile Computing (WiCom), 2009.
[9] P. Kyasanur and N. Vaidya, "Capacity of Multi-Channel Wireless Networks: Impact of Channels and Interfaces," ACM International Conference on Mobile Computing and Networking (MobiCom), 2005.
[10] M. Kodialam and T. Nandagopal, "Characterizing the Capacity Region in Multi-Radio Multi-Channel Wireless Mesh Networks," ACM International Conference on Mobile Computing and Networking (MobiCom), 2005.
[11] M. Alicherry, R. Bhatia, and L. Li, "Joint Channel Assignment and Routing for Throughput Optimization in Multi-radio Wireless Mesh Networks," ACM International Conference on Mobile Computing and Networking (MobiCom), 2005.
[12] A. Raniwala and T. Chiueh, "Architecture and Algorithms for an IEEE 802.11-based Multi-Channel Wireless Mesh Network," IEEE International Conference on Computer Communications (INFOCOM), 2005.
[13] K. Ramachandran, E. Belding-Royer, K. Almeroth, and M. Buddhikot, "Interference-Aware Channel Assignment in Multi-Radio Wireless Mesh Networks," IEEE International Conference on Computer Communications (INFOCOM), 2006.
[14] S. Roy, A. Das, R. Vijayakumar, H. Alazemi, and H. M. and E. Alotaibi, "Capacity Scaling with Multi-radio Mesh Networks," IEEE Workshop on Wireless Mesh Networks (WiMesh), 2005.
[15] K. Chen, Y. Xue, S. H. Shah, and K. Nahrstedt, "Understanding Bandwidth-Delay Product in Mobile Ad Hoc Networks," *Computer Communications*, vol. 27, pp. 923 – 934, 2003.
[16] P2 Mobile Technologies Limited, URL: http://www.p2mt.com/.
[17] Iperf, URL: http://sourceforge.net/projects/iperf/.
[18] W. Li, D. Zhang, J. Yang, and G. Xie, "On evaluating the differences of TCP and ICMP in network measurement," *Computer Communications*, vol. 30, pp. 428 – 439, 2007.
[19] G. R. Hiertz, S. Max, Y. Zang, T. Junge, and D. Denteneer, "IEEE 802.11s MAC Fundamentals," IEEE International Workshop on Enabling Technologies and Standards for Wireless Mesh Networking, 2007.
[20] GridFTP, URL: http://www.globus.org/toolkit/docs/latest-stable/gridftp/.
[21] bbftp, URL: http://doc.in2p3.fr/bbftp/.
[22] bbcp, URL: http://www.slac.stanford.edu/~abh/bbcp/.
[23] T. Dunigan, M. Mathis, and B. Tierney, "A TCP Tuning Daemon," ACM/IEEE Supercomputing Conference, 2002.
[24] K. K. Leung, T. E. Klein, C. F. Mooney, and M. Haner, "Methods to Improve TCP Throughput in Wireless Networks with High Delay Variability," IEEE Vehicular Technology Conference (VTC), 2004.
[25] Q. Pang, S. C. Liew, V. C. M. Leung, "Design of an Effective Loss-Distinguishable MAC Protocol for 802.11 WLAN," *IEEE Communications Letters*, vol. 9, no. 9, pp. 781-783, 2005.
[26] R. Ludwig and R. Katz, "The Eifel Algorithm: Making TCP Robust Against Spurious Retransmissions," *ACM Computer Communication Review*, vol. 30, no. 1, 2000.
[27] C. P. Fu and S. C. Liew, "TCP Veno: TCP Enhancement for Transmission over Wireless Access Networks," *IEEE J. on Selected Areas in Commun.*, vol. 21, no. 2, pp. 216-228, 2003.
[28] C. Jin, D. X. Wei, S. H. Low, G. Buhrmaster, J. Bunn, D. H. Choe, R. L. A. Cottrell, J. C. Doyle, W. Feng, O. Martin, H. Newman, F. Paganini, S. Ravot, S. Singh, "FAST TCP: from theory to experiments," *IEEE Network*, vol. 19, no. 1, pp. 4 – 11, 2005.
[29] D. X. Wei, C. Jin, S. H. Low, and S. Hegde, "FAST TCP: Motivation, Architecture, Algorithms, Performance," *IEEE/ACM Transactions on Networking*," vol. 14, no. 6, 2006.
[30] S. Kopparty, S. V. Krishnamurthy, M. Faloutsos, S. K. Tripathi, "Split TCP for Mobile Ad Hoc Networks," IEEE Global Telecommunications Conference (GLOBECOM), 2002.
[31] F. Xie, J. L. Hammond, and D. L. Noneaker, "Evaluation of a Split-Connection Mobile Transport Protocol," *Wireless Networks*, vol. 9, pp. 593 – 603, 2003.
[32] K. Liu, J. Y. B. Lee, "Mobile Accelerator: A New Approach to Improve TCP Performance in Mobile Data Networks," the 7th International Wireless Communications and Mobile Computing Conference (IWCMC), 2011.
[33] H. Balakrishnan, S. Seshan, E. Amir, and R. H. Katz, "Improving TCP/IP Performance over Wireless Networks," ACM International Conference on Mobile Computing and Networking (MobiCom), 1995.
[34] C. H. Ng, J. Chow, L. Trajkovic, "Performance Evaluation of TCP over WLAN 802.11 with the Snoop Performance Enhancing Proxy," OPNETWORK, 2002.
[35] V. Jacobson, R. Braden, D. Borman, "TCP Extensions for High Performance," https://rsync.tools.ietf.org/html/rfc1323, 1992.
[36] Microsoft Support, "Network connectivity fails when you try to use Windows Vista behind a firewall device," URL: http://support.microsoft.com/kb/934430/.

Ivan Wang-Hei Ho is currently a Research Assistant Professor at the Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hong Kong. He received the B.Eng. and M.Phil. degrees in Information Engineering from The Chinese University of Hong Kong, Hong Kong, in 2004 and 2006, respectively, and the Ph.D. degree in Electrical and Electronic Engineering from Imperial College London, UK, in 2010.

He worked on the Mobile Environmental Sensing System Across a Grid Environment (MESSAGE) project funded by EPSRC and Department for Transport, UK, and the International Technology Alliance (ITA) project funded by the US Army Research Laboratory and UK Ministry of Defence during his Ph.D. studies. In 2007, Ivan spent a summer working at the IBM T. J. Watson Research Center, Hawthorne, NY, USA. After his Ph.D. graduation, he was with the System Engineering Initiative at Imperial College London as a Postdoctoral Research Associate. In Sept 2010, he co-founded P2 Mobile Technologies Limited at Hong Kong Science Park and served as the Chief R&D Engineer.

His research interests are in wireless communications and networking, specifically in vehicular ad-hoc networks and intelligent transportation systems, wireless mesh networks, and capacity scaling and power control in wireless networks. His publications can be found at http://www.eie.polyu.edu.hk/~whho/.

Patrick P. Lam is currently the CEO of P2 Mobile Technologies Limited, Hong Kong. He has held senior managerial and architect roles in several telecommunications and software companies in Silicon Valley and Hong Kong since 1999. Between 1999 and 2002, he served as the Chief Engineer and Project Manager in Edgenet Communications Inc., Burlingame, California, USA, where he had led the company to successfully build the world's fastest H.323 VoIP gatekeeper at that time. He joined Hong Kong Applied Science and Technology Research Institute (ASTRI) in 2002 and co-founded the Wireless Communications Group under HKSAR's Innovation and Technology Fund. Prior to founding P2 Mobile Technologies, he was the Technical Software R&D Manager in the world's largest semi-conductor equipment manufacturer, ASM Pacific Technology Ltd.

He received his Ph.D. degree from The Chinese University of Hong Kong, Hong Kong, his Master degree from the University of British Columbia, Canada, and his Bachelor degree from the University of California, Irvine, USA. His core expertise includes 802.11, IP Mobility, Network Mobility, VoIP and various software technologies. He has published many research papers in these areas and is an US patent holder. He is also the co-author of the highly referenced Physical-layer Network Coding (PNC) technology.

Peter H. J. Chong received the B.Eng. (with distinction) in electrical engineering from the Technical University of Nova Scotia, Halifax, NS, Canada, in 1993, and the M.A.Sc. and Ph.D. degrees in electrical engineering from the University of British Columbia, Vancouver, BC, Canada, in 1996 and 2000, respectively.

Between July 2000 and January 2001, he worked in the Advanced Networks Division at Agilent Technologies Canada Inc., Vancouver, BC, Canada. From February 2001 to May 2002, he was with the Radio Communications Laboratory at Nokia Research Center, Helsinki, Finland, and was involved in research on WCDMA and standardization for HSDPA. During his stay in Finland, he has taught part of a graduate course in WCDMA at the Helsinki University of Technology, Helsinki, Finland. Since May 2002, he has been with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, where he is now an Associate Professor (Tenured). He is currently an Assistant Head of Division of Communication Engineering. He has visited Tohoku University, Japan, as a Visiting Scientist in 2010 and The Chinese University of Hong Kong (CUHK), Hong Kong, between 2011 and 2012. He is currently an Adjunct Associate Professor of CUHK.

He was a Technical Program Committee Chair for Mobility Conference 2005 and 2006, a Chair of Mobility Conference 2007 and 2008, a TPC Co-Chair of IEEE International Conference on Networks (ICON) 2012 and Co-Chair of International Conference on Information, Communications, and Signal Processing (ICICS) 2013. He served as a Guest Editor of Journal of Internet Technology in 2006, International Journal of Ad Hoc and Ubiquitous Computing in 2007 and lead Guest Editor of IEEE Communications Magazine in 2007 and IEEE Wireless Communications in 2011. He is an Editorial Board Member of Security and Communication Networks, International Journal of Wireless Communications and Networking, Wireless Sensor Network, and an Editor of Far East Journal of Electronics and Communications. His research interests are in the areas of mobile communications systems including radio resource management, multiple access, MANETs, multihop cellular networks and green radio networks.

Soung Chang Liew received his S.B., S.M., E.E., and Ph.D. Degrees from the Massachusetts Institute of Technology. From 1984 to 1988, he was at the MIT Laboratory for Information and Decision Systems, where he investigated Fiber-Optic Communications Networks. From March 1988 to July 1993, he was at Bellcore (now Telcordia), New Jersey, where he engaged in Broadband Network Research. He has been Professor at the Department of Information Engineering, the Chinese University of Hong Kong, since 1993. He is Adjunct Professor at Southeast University, China.

Prof. Liew's current research interests include wireless networks, Internet protocols, multimedia communications, and packet switch design. Prof. Liew's research group won the best paper awards in *IEEE MASS 2004* and *IEEE WLN 2004*. Separately, TCP Veno, a version of TCP to improve its performance over wireless networks proposed by Prof. Liew's research group, has been incorporated into a recent release of Linux OS. In addition, Prof. Liew initiated and built the first inter-university ATM network testbed in Hong Kong in 1993. More recently, Prof. Liew's research group pioneers the concept of *Physical-layer Network Coding* (PNC) for application in wireless networks.

Besides academic activities, Prof. Liew is also active in the industry. He co-founded two technology start-ups in Internet Software and has been serving as consultant to many companies and industrial organizations. He is currently a consultant for the Hong Kong Applied Science and Technology Research Institute (ASTRI), providing technical advice as well as help to formulate R&D directions and strategies in the areas of Wireless Internetworking, Applications, and Services.

Prof. Liew is the holder of seven US patents and a Fellow of IEEE, IEE, and HKIE. He is the recipient of the first Vice-Chancellor Exemplary Teaching Award at the Chinese University of Hong Kong. Publications of Prof. Liew can be found in www.ie.cuhk.edu.hk/soung.