On Accelerating Content Delivery in Mobile Networks

Tao Han, *Student Member, IEEE,* Nirwan Ansari, *Fellow, IEEE,* Mingquan Wu, *Member, IEEE,* and Heather Yu, *Member, IEEE*

***Abstract*—Owing to the imminent fixed mobile convergence, Internet applications are frequently accessed through mobile devices. Given limited bandwidth and unreliable wireless channels, content delivery in mobile networks usually experiences long delay. To accelerate content delivery in mobile networks, many solutions have been proposed. In this paper, we present a comprehensive survey of most relevant research activities for content delivery acceleration in mobile networks. We first investigate the live network measurements, and identify the network obstacles that dominate the content delivery delays. Then, we classify existing content delivery acceleration solutions in mobile networks into three categories: mobile system evolution, content and network optimization, and mobile data offloading, and provide an overview of available solutions in each category. Finally, we survey the content delivery acceleration solutions tailored for web content delivery and multimedia delivery. For web content delivery acceleration, we overview existing web content delivery systems and summarize their features. For multimedia delivery acceleration, we focus on accelerating HTTP-based adaptive streaming while briefly review other multimedia delivery acceleration solutions. This paper presents a timely survey on content delivery acceleration in mobile networks, and provides a comprehensive reference for further research in this field.**

***Index Terms*—Content delivery acceleration, mobile networks optimization, mobile system evolution, content adaptation and optimization, mobile data offloading.**

I. Introduction

CONTENT delivery acceleration, whose market is forecasted to grow to $ 5.5 billion in 2015 [1], attracts tremendous research efforts from both industry and academia [2]. With the rapid development of radio access techniques and mobile devices, Internet applications are gradually moved to mobile networks. As shown in Fig. 1, mobile data traffic is forecasted to increase exponentially, and mobile web and video applications are the major mobile data generators which account for 20% and 70.5% of the total data traffic, respectively [3]. Mobile web applications consume low bandwidth, but are sensitive to network latency. Subscribers usually expect a page to be loaded in less than two seconds, and 40% of subscribers wait for no more than 3 seconds before leaving the web sites [4]. Thus, for a content provider, a one second delay in page load time can result in lost conversions, fewer page views, and a decrease in customer satisfaction [5].

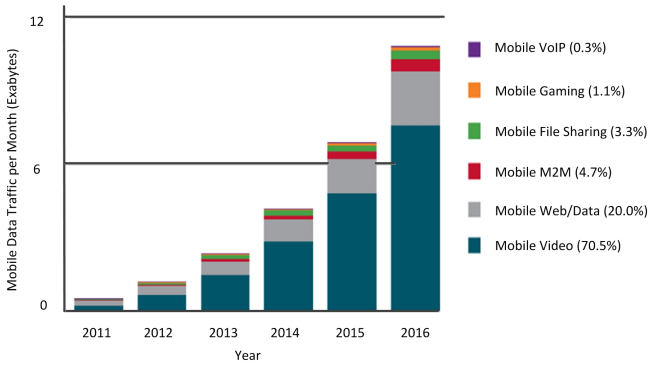


Fig. 1. Mobile data traffic share [3].

Mobile video applications generate the largest wireless data traffic volume. Different video applications behave differently in term of bandwidth consumption. For example, applications such as Netflix and Hulu adopt adaptive HTTP streaming which adjusts the bit rates according to the network conditions. Adaptive HTTP streaming usually consumes as much bandwidth as possible to maintain the best possible quality of the video. Video applications like YouTube, however, behave differently. They usually start at the lowest possible bit rate, and allow the subscribers to select the bit rates. Despite the difference on bandwidth usages, both categories of video applications are bandwidth consuming, and their bandwidth consumption grows rapidly owing to the availability of the high-definition video and the increased screen size of the mobile devices. Therefore, accelerating mobile web and video content delivery is crucial to enhance the quality of experience (QoE) of the subscribers in mobile networks.

Fig. 2 illustrates the world wide mobile subscriptions by technology. GSM/EDGE enabled mobile networks attract the largest number of subscribers and will continue to share a large portion of the total number of mobile subscribers. The number of subscribers of WCDMA/HSPA technology grows rapidly, and will lead the market by 2017. LTE is currently being deployed and will be used by a small portion of the subscribers [5]. Therefore, most of the exponentially increased data traffic will inject into the mobile networks powered by GSM/EDGE or WCDMA/HSPA technologies. Suffering from the dramatic data traffic increases, the content delivery delay can be arbitrarily long, thus degrading subscribers' QoE. Hence, network

Manuscript received 16 June 2011; revised 17 January 2012 and 23 June 2012.

T. Han, and N. Ansari are with the Advanced Networking Lab., Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, NJ, 07102 USA (e-mail: {th36, nirwan.ansari}@njit.edu).

M. Wu and H. Yu are with Huawei Technologies, USA (e-mail: {Mingquan.Wu, heatheryu}@huawei.com).

Digital Object Identifier 10.1109/SURV.2012.100412.00094

1553-877X/13/$31.00 © 2013 IEEE

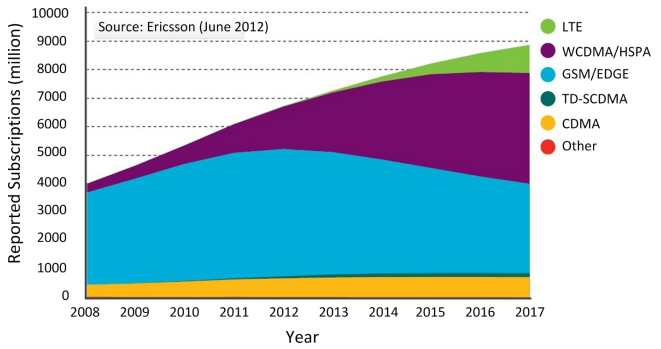


Fig. 2. Mobile subscriptions by technology, 2008-2017 [5].

optimization and content delivery acceleration solutions are urgently needed in GSM/EDGE or WCDMA/HSPA mobile networks. Therefore, we focus the scope of the survey on mobile networks powered by GSM/EDGE or WCDMA/HSPA technologies.

To understand the performance of mobile networks, many measurement studies have been presented. These studies unveil the obstacles that delay content delivery in mobile networks, and shed lights on the research directions for enhancing the performance of mobile networks. Noticing the shortcomings of mobile networks, many solutions have been proposed to reduce content delivery latency and enhance subscribers' QoE in mobile networks. Fig. 3 shows the classification hierarchy of available content delivery acceleration solutions. We classify these solutions into three categories, the mobile system evolution, the content and network optimization, and the mobile data offloading. Within each category, the techniques are further classified. Mobile communication system evolution is one of the major solutions to address the problem with mobile networks. On the one hand, to meet the increasing demands for mobile data services, the 3rd Generation Partnership Project (3GPP) has established evolution plans to enhance the performances of mobile communication systems. 3GPP LTE (Long Term Evolution) Advanced is a mobile communication standard for next generation mobile communication system featured with high speed and low latency. LTE Advanced networks adopt multi-input-multi-output (MIMO) and orthogonal frequency-division multiple access (OFDMA) to enhance the capacity as well as the reliability of wireless links, introduces the EPS (Evolved Packet System) [6] to reduce the amount of protocol related processing, and integrates cognitive radio techniques to expand the available bandwidth in the system. On the other hand, mobile networks and content delivery networks are being integrated to provide end-to-end acceleration for mobile content delivery [7].

The content and network optimization techniques are further classified into three categories based on their application domains. The first category pertains to the content domain techniques including caching, data redundancy elimination, prefetching, and data compression. These techniques aim to reduce the traffic volume over mobile networks, thus reducing the network congestion and accelerating content delivery. The second category refers to the techniques applied in the network domain. These techniques include the the handover optimization, the queue management techniques, network coding, the TCP optimization, and the session layer optimization. The network domain techniques optimize the operation of mobile networks and communication protocols, and thus enhance network performance. The third category includes the cross domain techniques such as the content adaptation techniques and the protocol adaptation techniques. Content adaptation is to adjust the original content according to the mobile network conditions and the characteristics of mobile devices. Content adaptation can efficiently reduce the data volume over mobile networks and accelerate the content delivery. Protocol adaption is to optimize the communication protocols according to the application behaviors. It reduces the network chattiness, and thus reduces the content delivery delay.

The significant data traffic increase may congest the mobile network, and lead to long delay in content deliveries. Offloading data traffic from congested mobile networks is a promising method to reduce network congestion. Mobile data offloading techniques include two perspectives. The first one is to directly offload mobile data to high speed networks, e.g., WiFi. The second one is network aggregation which allows subscribers simultaneously utilizing their multiple radio interfaces, e.g., 3G, WiFi, and Bluetooth, to retrieve the content. Mobile data offloading techniques reduce the pressure on mobile networks in term of data volume, thus enhancing network performance in term of content delivery.

The rest of the paper is organized as follows. Section II analyzes the live network measurements and identifies the dominant obstacles that delay the content delivery in mobile networks. Section III discusses the latency reduction achieved by mobile system evolution. Section IV provides an overview of content and network optimization techniques. Section V presents mobile data traffic offloading techniques. Section VI discusses the web content delivery acceleration systems for mobile networks. Section VII presents multimedia delivery acceleration solutions. Section VIII concludes the paper and discusses several open issues related to content delivery acceleration in mobile networks.

II. Mobile Network Measurements

Understanding the performance of mobile networks and identifying the performance bottlenecks are the first step toward accelerating content delivery in mobile networks. In this section, we overview the studies on mobile networks measurements and generalize the dominant factors that deteriorate network performance with respect to the content delivery delay.

A. Packets Retransmission

To alleviate the impact of wireless errors on network performance, 3G mobile networks adopt two-layer retransmission mechanisms: MAC (Media Access Control) layer hybrid-ARQ (Automatic Repeat-reQuest) and RLC (Radio Link Control) layer ARQ [8]. Hybrid-ARQ located in NodeB targets fast retransmission and provides feedback on the decoding attempts to the transmitter after each transmission. The excessive feedbacks introduce additional cost to the mobile system. To keep a reasonable feedback overhead, hybrid-ARQ does

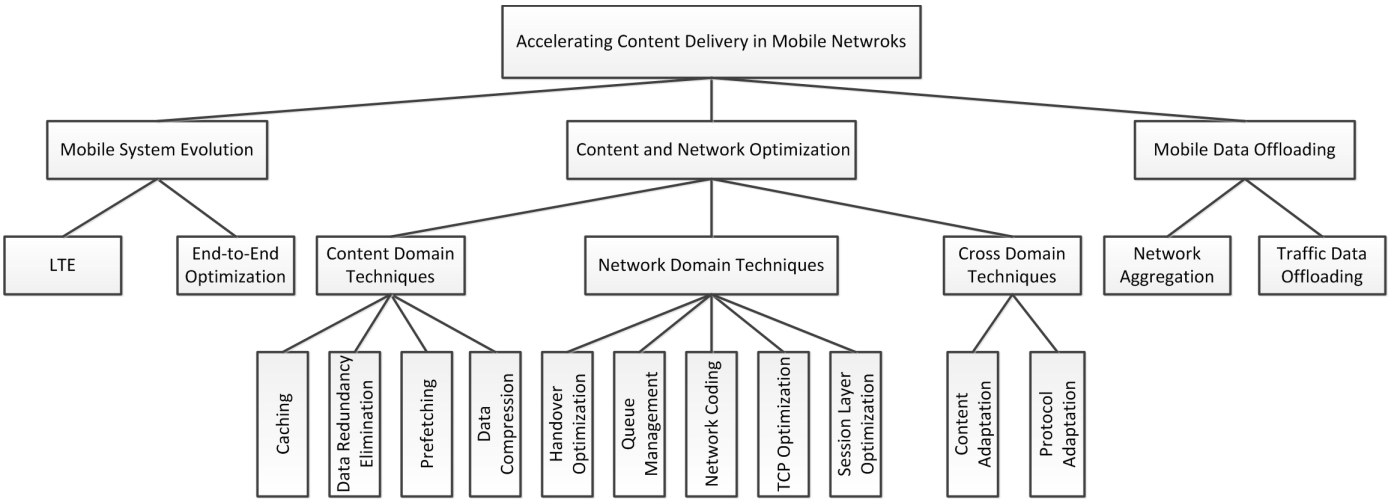


Fig. 3. Classification hierarchy of content delivery acceleration solutions in mobile networks.

not provide feedback on every decoding attempt that results in a hybrid-ARQ residual error. RLC layer ARQ located in the RNC (Radio Network Controller) requires relatively infrequently RLC status reports, and can achieve low error rate with small cost. However, RLC layer ARQ takes more packet recovery time than hybrid-ARQ. The two-layer retransmission architecture achieves a fast retransmission attributed to hybrid-ARQ and a reliable packet delivery facilitated by RLC layer ARQ.

However, these retransmission mechanisms introduce delay spikes, and around 15% of the packets over the UMTS (Universal Mobile Telecommunications System) network are affected by the delay spikes [9]. One reason for the delay spike is the out of order delivery of transport blocks, PDUs (Protocol Data Units). Since the hybrid-ARQ processes operate independently, PDUs may be delivered out of order. In this case, the RLC layer ARQ, which assures in-sequence delivery, has to buffer all the out of order PDUs and to reorder them, thus resulting in the delay spike. Another reason of the delay spike is the residual errors of the hybrid-ARQ. To assure reliable delivery, RLC layer retransmissions are scheduled when hybrid-ARQ processes fail to deliver the data units. RLC layer ARQ consume more time than hybrid-ARQ processes, and cause the delay spikes. Such delay spikes increase the RTT of mobile networks and result in a long delay of a significant IP-packets service time [10]. The long packet delay sometimes causes TCP RTO (Retransmission Timeout) that further deteriorates network performance [11].

B. Queuing in Mobile Core Networks

In a mobile core network, data services are supported by the packet switch domain, which consists of SGSN (Serving GPRS Support Node) and GGSN (Gateway GPRS Support Node). SGSN performs mobility management, logical link control, and data units routing while GGSN acts as an IP gateway which connects the mobile core network with the WAN (Wide Area Network). Since the mobile core network connects the high speed Internet with low speed radio network, the queuing delay in the mobile core network is unavoidable. The queuing delay is exacerbated by the large sending buffer in SGSN. To accommodate the time-varying wireless channel and improve the channel utilization, SGSN is usually equipped with large buffers, which result in long queuing delay when the wireless channel rate is low due to the bad channel condition or user mobility [12]. In addition, heavy traffic loads of the cell negatively impact the queuing delay [13]. As a result, the queuing delay in SGSN dominates 3G core network latency [14]. The excessive queuing delay leads to RTT inflation, retransmission timer inflation, SYN timeout, and higher recovery time, thus prolonging the packet delivery time.

C. Network Asymmetry

As compared to the downlink, the uplink of mobile networks has smaller channel rates and is prone to wireless errors [15]. The reasons for the asymmetry are multifold. Unlike the downlink channel, the uplink channel is non-orthogonal and subject to interference between uplink transmissions. Also, with limited processing ability and transmission power, the high order modulation is less useful. In addition, the uplink scheduler and the transmission buffers are located in different places: the former is located in NodeB while the later in UE (User Equipment). Such separation requires UE signal buffer status information to the scheduler since the wireless schedulers are very sensitive to the buffer status (application rates) [16]. The additional signalling processes may result in inefficient scheduling. Because of the weaknesses, the uplink performance limits the RTT of the networks [11], and the UE with larger uplink traffic experiences longer RTT [17].

D. Queue Management

Owing to the differences between wireline networks and mobile networks, the queue management scheme designed for wireline networks, e.g., RED (Random Early Detection), may face some problems in mobile networks [18]. RED monitors the average queue length and drops packets based on statistical probabilities. If the average queue length is smaller than a predefined threshold, all incoming packets

are accepted. Otherwise, the incoming packets will be drop base on statistical probabilities which are increasing as the queue grows. When the queue is full, all incoming packets will be dropped [19]. One problem is that a static minimum threshold [1] cannot effectively maintains the queue size. A static minimum threshold often fails to reflect the capacity of the time varying wireless channel. Moreover, the mobile network has a larger queue draining latency due to its lower bandwidth. As a result, if the mobile network experiences a temporal bandwidth shrink, a queue will be built up, and is not easily drained. Another problem is related to the probabilistic discarding. The probabilistic discarding may lead to a delayed congestion feedback. Considering the large queue draining delay, the delayed feedback can result in over-buffering and excessive queuing delay. In addition, mobile networks keep a per user queue, and thus the statistical multiplexing in the queue is lower since the queue is shared by at most $2-4$ flows. The probability that the discarded packets are from the same connection is higher. Therefore, the connections that are not granted fair share of the bandwidth will experience longer delay.

E. First Packet Delay

In 3G networks, a subscriber's first packet tends to experience high jitter due to the delay in state transition and the delay related to the network registration and resource allocation [20]. UEs may be in different RRC (Radio Resource Contorl) states, which are shown in Fig. 4. UEs are either in the idle mode or connected mode when they are powered on. When UEs are in the connected mode, they may be in four different states: CELL_DCH (Cell Dedicated Channel), CELL_FACH (Cell Forward Access Channel), CELL_PCH (Cell Page channel), and URA_PCH (UTRAN Registration Area Paging Channel) [21]. The state transitions between different states have a direct impact on the latency. For instance, it usually takes $2-4$ seconds to transit from $CELL\_FACH$ to $CELL\_DCH$ since it requires to set up the DCH (Dedicated Channel) channel [11]. As a result, the first packet tends to experience longer delay. Another reason of the first packet delay is related to the network registration and the resource allocation. To access the packet switch services, UEs have to experience a three-phase activation process. They are GPRS Attach, PDP Content Activation and RAB Establishment, and Register with IMS [22]. After this process, the wireless resources are allocated to the UE, and then the UE can access the packet switch services. Such activation process introduces additional latency of the first packet.

F. TCP Flaws

TCP is one of the core protocols of the Internet protocol suites. Many popular applications such as web services, http-based streaming, e-mail and file transfer rely on TCP to provide reliable point to point communications. Therefore, the performance of TCP has a significant impact on the service delivery latency over wireless networks [23]. In this subsection, we discuss two TCP flaws: TCP slow start and TCP ACK compression, and their impacts on delivery latency.

[1]If the queue size is larger than the minimum threshold, the incoming packets will be dropped according to a predefined probability.

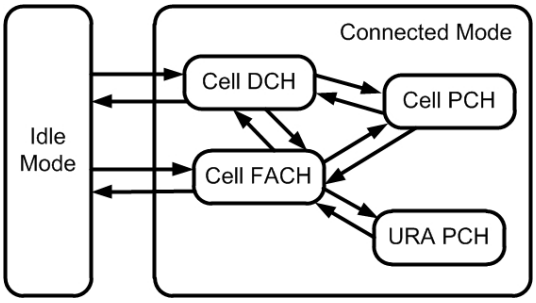


Fig. 4. The RRC states in UMTS [21].

1) TCP Slow Start: TCP utilizes the slow start and the congestion avoidance to probe the available bandwidth. At the beginning of one connection, the TCP sender sends the data units according to a predefined CWND (Congestion WiNDow), and exponentially increases the CWND at every ACK (ACKnowledgement) until reaching the predefined threshold. Then, TCP enters the congestion avoidance phase. The slow start of TCP has a significant impact on the user perceived service response time [24]. At the start up phase, the bandwidth is not fully utilized, and it takes several RTTs to ramp the congestion window up to the link BDP (Bandwidth Delay Product) [25]. Moreover, most of the TCP flows generated by popular mobile services, e.g., web service, are short lived and only experience the slow start phase [26]. Therefore, the user perceived network throughput is limited by the initial CWND rather than the network capacity. Such limitation deteriorates the service latency.

2) TCP ACK Compression: TCP ACK compression is a well known problem in wireless networks [27]. Owing to the link asymmetry and in-order delivery, several ACKs can be compressed. ACK compression disturbs the synchronization between the TCP sender and the receiver, and worsens the network congestion, therefore increasing the service response time over mobile networks [10].

G. Application Misbehavior

The applications optimized for Internet may misbehave in the wireless environment. One of the misbehavior is the TCP concurrency which refers to the case that the application opens several TCP connections with the server simultaneously. TCP concurrency is efficient in wireline networks since it reduces the impact of TCP slow start and the delay caused by in-sequence fetching. However, using multiple TCP connections has several flaws [28] over wireless networks. First, the simultaneous TCP connections often cause the so-called self congestion, and delay the new TCP connection setup [24]. Although Huang *et al.* [29] showed that using multiple TCP connections in wireless networks improves the web page download time by 30%, the study is under the assumption that the network condition (the RTT) is unchanged during the experiment. The real world measurement showed that the RTT increases as the number of simultaneous TCP connection increases [17]. The RTT inflation has a negative impact on the web page download time, and results in degraded network

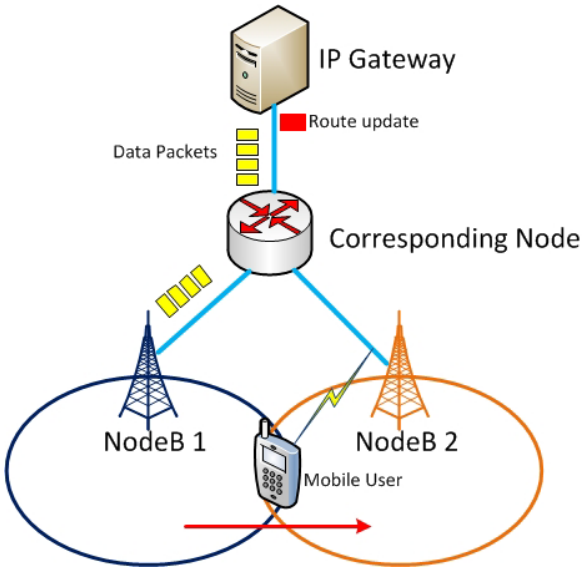


Fig. 5. Packets loss in the handover process.

performance. Second, establishing a TCP connection, which requires the three way hand shake, introduces significant latency. Third, the protocol control overhead is high and consequently degrades the performance. Thus, the number of concurrent TCP connections should be optimized according to the wireless network condition; otherwise, it will introduce additional delay rather than improvements.

H. Mobile Devices

Because of the limited capability, the mobile devices themselves are found to be one of the limiting factors for mobile applications. One of the limitations is the low transmission power of the mobile device that results in a relatively unreliable uplink and TCP misbehavior, e.g., ACK compression, thus increasing the service response time. Another limitation is the limited computing ability and small buffer size. With the limited computing ability, UE requires long time to process the incoming data, especially for the multimedia content. Given the small buffer size, the receive buffer will be filled quickly. Thus, the TCP sender will pause sending the data since the receive buffer is not available. This results in a long period of TCP idle during the data transmission, thus significantly impacting the service response time [29].

I. User Mobility

User mobility, which often triggers the handoff process, impacts network latency by introducing the handover delay. As a result, mobile users experience far worse performance than stationary users [30]. Network layer handover latency, which is caused by network registration delay and route update delay, is one of the limitations for IP-based mobile networks [31]. In addition, the handover processes can result in high packet losses [32]. As shown in Fig. 5, if the data packets arrive at the corresponding point earlier than the route update information, they will be forwarded through the old path to NodeB 1. Since the mobile user is already attached to NodeB 2, data packets sent through NodeB 1 will be lost. These packet losses may result in excessive delays.

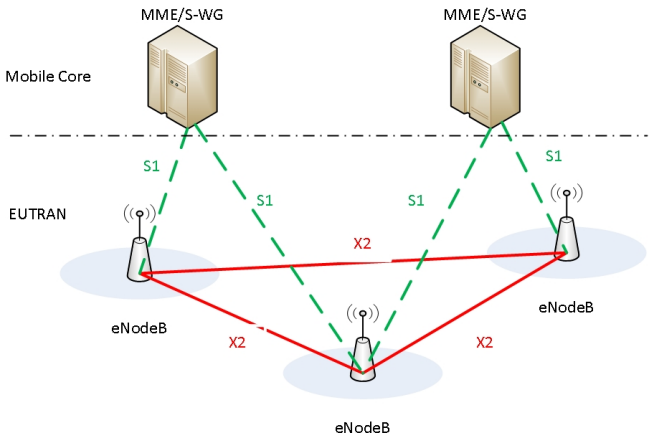


Fig. 6. Overall System Architecture [6].

III. Mobile System Evolution

To reduce network latency and secure subscribers' QoE, mobile communication systems are enhanced from two perspectives. On the one hand, a new radio access network architecture named EUTRAN has been adopted in 3GPP LTE Advanced. EUTRAN adopts a flat architecture which simplifies signaling processes and reduces network latency. On the other hand, mobile networks and content delivery networks are being integrated to accelerate content delivery in mobile networks.

A. EUTRAN

As shown in Fig. 6, EUTRAN applies a flat architecture which allows eNodeBs communicate with each other through X2 interfaces. EUTRAN eliminates RNC, and moves its functions into eNodeB. By simplifying the system architecture, EUTRAN reduces the overall amount of protocol-related processing, and thus reduces the latency. In the following, we discuss the latency reduction in both the control plane and user plane in EUTRAN.

1) Control Plane Latency Reduction: EUTRAN adopts a simple RRC state machine and limits its states to two: RRC_IDLE and RRC_CONNECTED as shown in Fig. 7. In the RRC_IDLE state, there is no connection between eNodeB and UE. The state switching from RCC_IDLE to RCC_CONNECTED is triggered by either the service activation from UEs or paging messages from eNodeBs. In the RRC_CONNECTED states, there is an active connection between UE and eNodeB, and data or signaling message can be exchanged over wireless channels. If there is no service activity for a certain duration, UE returns to the RRC_IDLE state. As compared to RRC states in UMTS as shown in Fig. 4, the simple RRC state machine in EUTRAN helps reduce the first packet delay as discussed in Section II. Mohan *et al.* [33] compares control plane latency in UMTS and LTE, and showed that the latency of the mobile originated call setup from the idle state has been improved from 1717 ms in UMTS to 188 ms in LTE. This significant latency reduction improves the responsiveness of mobile networks.

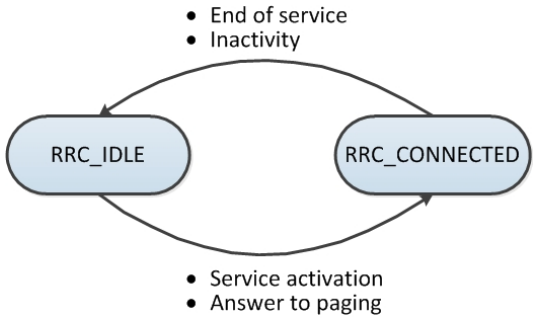


Fig. 7. The RRC states in EPS [34].

2) User Plane Latency Reduction: As shown in Fig. 8, radio access networks of EPS (EUTRAN) has eliminated RNC, and relocates its functions to eNodeB. In EUTRAN, compression and ciphering functions are supported by the PDCP (Packet Data Convergence Protocol) layer at eNodeB. In addition, both hybrid-ARQ and RLC layer ARQ are accommodated in the eNodeB. Therefore, in EUTRAN, only one buffer of header compressed and ciphered IP packets is required for data processing. This system architecture helps reduce latency in the user plane from two aspects. First, this architecture reduces the queuing delay in radio access networks. In UMTS, data processing requires two separated queues at RNC and NodeB, respectively. Hence, a flow control mechanism is required in order to maintain an optimal queue size at NodeB. The flow control mechanism between RNC and NodeB may introduce additional queuing delay. By merging functions of RNC and NodeB into eNodeB, only one buffer is required for data processing in EUTRAN. Therefore, the queuing delay is reduced. Second, this architecture reduces the delay caused by handover processes. As mentioned in Section II, excessive delays may be introduced by handover processes in 3G mobile networks. EUTRAN introduces a new inter-eNodeB interface, X2, to reduce the delay caused by handover processes. The X2 interface enables a flat architecture in EUTRN which accelerates the network registration procedure during handover. In addition, X2 interfaces allow data buffer to be forwarded between source and target eNodeBs, thus reducing the probability of packet loss during handover processes [34].

B. Integrating Mobile Networks and CDN

Mobile networks which are originally designed for voice and data communications are not optimized for content delivery. Therefore, the performance of mobile networks in term of content delivery cannot be guaranteed. Integrating mobile networks and CDN provides end-to-end solutions for content delivery in mobile networks and enhances network performance. Ericsson and Akamai proposed the mobile cloud accelerator which reserves a portion of network bandwidth for the premium content. Therefore, the content with higher priority can avoid the delay caused by network congestion, and be delivered faster. In addition, Blumofe *et al.* [35] proposed to deploy caching proxy into mobile networks to accelerate content delivery.

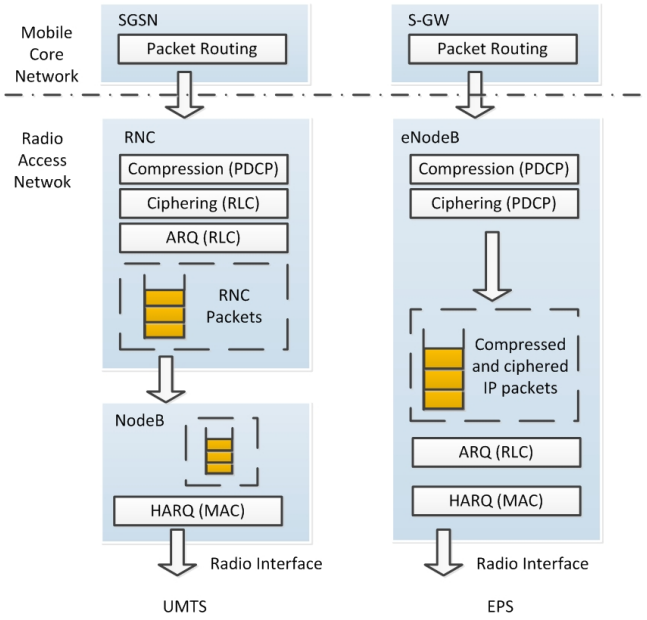


Fig. 8. UMTS and EPS downlink user plane handling comparison [34].

IV. Content and Network Optimization

In this section, we present an overview of available content and network optimization techniques which are classified into content domain techniques, network domain techniques, and cross domain techniques.

A. Content Domain Techniques

Wireless links usually have constrained bandwidth that leads to an excessive queuing delay in mobile core networks. Reducing the traffic volume over mobile networks can accelerate the wireless transmission, thus alleviating the queuing delay. Content domain techniques aim to reduce the data volume over mobile networks. They include caching, data redundancy elimination, prefetching, and data compression.

1) Caching: Caching stores copies of frequently requested content in subscribers' local cache or in the cache proxy located at the network edge. Instead of being responded by the original content server, subscribers' requests are responded by their local cache or the cache proxy close to them. In this way, caching effectively reduces network traffic and the number of RTTs required to download content, thus improving the content delivery performance in term of the service response time. The performance of the caching scheme is determined by the hit rate of the cache; the higher the hit rate, the better the performance. To maximize the hit rate, Chakravorty *et al.* [28] presented a novel caching scheme that utilizes content hash to eliminate the redundant caching. The caching scheme indexes the objects by using content hash, and maps the URLs to the respective hash key. Since the same objects have the identical hash value, they are only cached for one time even though they may be pointed to by different URLs. In dynamically generated web pages, mapping multiple URLs to the same content is commonplace. Therefore, the content hash based caching scheme can significantly improve the hit rate of the client cache.

2) Data Redundancy Elimination: Data redundancy elimination techniques have been proven to be an efficient way to reduce the traffic load on bandwidth constraint networks [36]. The redundancy elimination algorithms rely on the caches deployed at both end of the link. When there is a packet, the ingress nodes find the common sequences of the bytes in the packets in their cache and replace them with a fixed size pointer. Receiving the pointer, the egress nodes decode the content from their local caches. However, the wireless loss can lead to unsynchronized caches between the sender and the receiver, and results in incorrect decoding of the content. To address this problem, the informed marking scheme was proposed to detect the packet losses and to prevent redundancy elimination algorithms from using them for encodings [37]. With the informed marking, redundancy elimination algorithms can save more than 60% of the bandwidth, thus reducing the queuing delay in the mobile core networks.

3) Prefetching: Prefetching techniques hide network latency from the users by predicting the users' next requests, and pre-retrieving the corresponding contents. The prefetching system usually consists of two modules: a prediction module that predicts the users' next requests and a pre-retrieving module that processes the prefetching. The prediction algorithms can be classified into two major categories: history-based prediction algorithms and content-based prediction algorithms [38]. The history-based prediction algorithms predict the future requests based on the observed pattern of past requests. There are two main techniques for the history-based prediction algorithms: Markov precess models [39], and data mining techniques [40]. The content based prediction algorithms focus on the content of the web pages that have been requested, and identify the contents that are of particular interests to the users [41]. The recommendations from the prediction module are stored in a hint list, which consists of a set of URLs that are likely requested by the user in a near future. The pre-retrieving module fetches the URLs in the hint list when the bandwidth is available. Khemmarat *et al.* [42] proposed a video prefetching system which consists of a search result-based prefetching scheme and a recommendation-aware prefetching scheme. The search result-based prefetching scheme utilizes the video search results provided by the video sharing sites such as Youtube and Youku as hints, while recommendation-aware prefetching scheme is to prefetch the video prefix in the recommendation list provided by the video sharing sites.

4) Data Compression: Data compression techniques accelerate content delivery by reducing the data volume generated by applications. For web services, data compression algorithms can be classified into two categories: the intra web page compression and the inter web pages compression. The intra web page compression techniques reduce the data redundancies within the web page. The intra web pages compression can be further classified as lossless compression and lossy compression. The former applies to the text or binary based content such as HTML files and Javascripts. The later usually applies to the image based contents like pictures and videos, and it trades the image quality for the image size.

The inter web pages compression minimizes the redundancies between the web pages. Delta encoding is one of the inter web pages compression techniques. It identifies the difference between the new web page and the old one, and sends only the difference to the web clients. Using delta encoding, the web clients avoid downloading the whole web page very time it is accessed. Therefore, delta encoding can significantly reduce the amount of traffic over the network, thus accelerating the web services [43].

B. Network Domain Techniques

Network domain techniques, which are to optimize mobile networks and communication protocols, can be classified into five categories: the handover optimization, the queue management techniques, network coding, the TCP optimization, and the session layer optimization.

1) Handover Optimization Techniques: As shown in the network measurement studies, handover delays may cause excessive packet losses and degrade network performance in term of content delivery. IP soft handover can reduce the handover latency and the packet loss during handover processes. It reduces the handover latency by providing the mobile hosts the information about the new access point and the associated subnet prefix information before they switch to the new access point [44]. Thus, the subscribers can execute the network registration and address resolution prior to the switch. In addition, as compared to hard handover, it reduces the packet loss because it does not disconnect from the previous access point until the route information is updated in the corresponding node. Nurvitadhi *et al.* [32] proposed adaptive semi-soft handover that further improves the latency performance by optimizing the tune-in time [2] based on the network conditions.

2) Queue Management Techniques: As discussed in the network measurement studies, queue management schemes designed for Internet face several problems in mobile networks. By addressing these problems, the queue management schemes tailored for mobility can enhance network performance. Sagfors *et al.* [18] proposed to exploit the knowledge about the time varying link's capacity, and to dynamically set the minimum threshold of RED to reflect the network conditions. In addition, they proposed to apply deterministic dropping strategies rather than probabilistic dropping strategies. In the deterministic dropping policy, the congestion is signaled as soon as it is detected by dropping a single packet. The dropping policy incorporates the discard prevention counter algorithm to avoid multiple losses from the same TCP connection.

3) Network Coding Techniques: Network coding has emerged as promising techniques to enhance the robustness and effectiveness of data transmission over lossy wireless networks. In the following, we discuss two potential applications of network coding that can improve mobile network performance.

Improve HARQ Efficiency: HARQ is adopted by 3G and 4G mobile networks to enhance the reliability of wireless links. The signaling overhead of HARQ may introduce additional latency into mobile networks. Lang *et al.* [45] proposed network coded HARQ (NC-HARQ) which integrates

[2]The tune-in time is the time when the mobile host switch from the old access point to the new one.

network coding into HARQ to increase the efficiency of HARQ in term of throughput, thus reducing the signaling overhead. NC-HARQ applies to the scenarios where there are consecutive erroneous packets. For example, if there are two consecutive erroneous packets, instead of retransmitting them individually, NC-HARQ codes the packets using XOR into one retransmission packet. In this case, NC-HARQ only requires three transmissions to handle two consecutive erroneous packets while the original HARQ mechanism requires four transmissions. Therefore, NC-HARQ enhances the HARQ throughput.

Enhance TCP performance: Most of popular Internet applications, e.g., web services and video streaming, are based on TCP to achieve reliable data transfer. However, TCP does not perform well in mobile networks because of unreliable wireless links. Therefore, enhancing TCP performance in mobile networks is crucial for accelerating mobile content delivery. Network coding is one of promising technologies that may enhance network performance in lossy networks. Therefore, integrating network coding into TCP may enhance TCP performance in lossy networks. Sundararajan *et al.* [46] proposed TCP-Network Coding (TCP-NC) which takes advantages of network coding to mask losses from the congestion control algorithm and achieves effective congestion control over lossy wireless networks. As shown in Fig. 9, TCP-NC implements a network coding layer between the internet protocol layer and the TCP layer. The network coding layer accepts packets from the TCP layer and buffers them into encoding buffer until they are acknowledged by the receiver. The sender sends a linear combination of the packets in the encoding buffer. Upon receiving a linear combination, the network coding layer at the receiver side finds out which packets have been newly seen because of the new arrival packets. The seen packets define an ordering of the degree of freedom [46] (i.e., a linear combination that reveals one unit of new information) which is consistent with the packet sequence number. The receiver ACKs the degree of freedom to the sender that allows the TCP congestion window at the sender side to advance. From the TCP sender's view, it appears that the transmitted packets wait in a fictitious queue until all the degree of the freedom is acknowledged. Therefore, TCP-NC translates the lossiness of the wireless link into an additional queuing delay. Thus, TCP-NC adopts TCP-Vegas which is a delay-triggered congestion control mechanism at its TCP layer. Kim *et al.* [47] evaluated the performance of TCP-NC in lossy wireless networks, and showed that TCP-NC is able to increase window size fast and maintain a large window size despite of the wireless losses.

4) TCP Optimization: Beside network coding, other solutions have been proposed to optimize TCP performance in mobile networks. These solutions can be classified into two categories. The first category is to design transmission control mechanisms to maximize the utilization of the available bandwidth. The second category is to design ACK mechanisms to provide accurate feedback on network conditions to the TCP sender.

Transmission Control Mechanisms: To speed up TCP connections, several TCP variants have been introduced. Zhang and Qiu [48] proposed TCP/SPAND, which significantly reduces the latency for short transfers. TCP/SPAND utilizes the shared network performance information to estimate the fair share of network resource. Then, based on both the estimation and the transfer data size, the TCP sender will determine an optimal CWND size, which minimizes the number of RTTs required for the transfer. However, the optimized CWND size can be potentially large. To limit the maximum burstiness of the outgoing traffic caused by the large CWND, TCP/SPAND applies a leaky-bucket based pacing scheme to smoothly sending out the packets in the initial CWND.

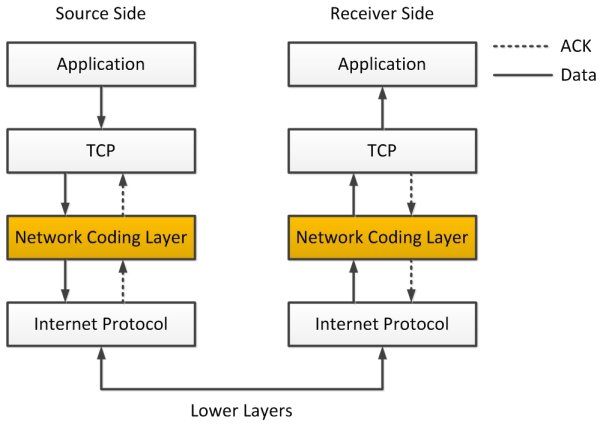


Fig. 9. New network coding layer in the protocol stack [46].

Chakravorty and Pratt [26] designed TCP CWND Clamping to eliminate the latency caused by TCP slow start. The TCP variant fixes the initial CWND to an estimation of the BDP (Bandwidth Delay Products). After sending out the packets in the initial CWND, the TCP CWND Clamping goes into a self-clocking state in which it clocks out one segment each time it receives an ACK for an equivalent amount of data from the receiver. Such design reduces the queuing delay since it limites the cwnd to BDP, and enables quick recovery from loss by limiting the data over the link. Goff *et al.* [49] proposed Freeze-TCP to prevent the TCP sender from dropping the CWND during the handoff process. In Freeze-TCP, the mobile user monitors the signal strength, detects the handoff, and predicts a temporary disconnection. Then, the mobile user signals the TCP sender to freeze the TCP flow to prevent the reduction of CWND due to excessive packet losses during the handoff process.

Rodriguez and Fridman [50] proposed TCP connection sharing which takes advantages of the concurrent TCP connections to estimate the RTT and congestion window size. The enhanced TCP stack caches the RTTs and congestion window sizes of current or recently expired TCP connections, according to which, TCP estimates the initial CWND of new connections to the same mobile host. In this way, the starting parameters can be effectively estimated, thus reducing the latency incurred by the TCP slow start phase.

Han *et al.* [51] designed a new TCP algorithm, TCP-ME (Mobile Edge), to accelerate content delivery in mobile networks. Considering the QoS (Quality of Service) mechanisms of mobile networks, TCP-ME is designed to differentiate the packets loss caused by wireless errors, traffic

conditioning mechanism of mobile core networks, and Internet congestion, and then to react to the packet loss accordingly. To detect wireless errors, the authors suggested to mark the ACK (Acknowledge) packets in the uplink direction at the base station, and the marking threshold is a function of the instantaneous downlink queue and the number of consecutive HARQ retransmissions. Inspired by TCP-New Jersey [52], the ECN mechanism is modified with deterministic marking to detect Internet congestion. The packet loss caused by traffic conditioners of mobile networks is detected if the incoming DUPACK is not marked. TCP-ME adapts the inter-packets intervals when the packet loss is caused by wireless errors or admission control mechanism. If the packet loss is due to Internet congestion, TCP-ME applies TCP-New Reno's congestion window adaptation algorithm. Simulation results show that TCP-ME can improve about 80% web service response time in mobile networks.

ACK Mechanisms: As discussed in the measure studies, in mobile networks, the hybrid ARQ and RLC retransmission can cause the increased delay and the rate variability, which, together with the requirement of in-order delivery, translate to ACK compression. The bursty ACK arrivals lead to release of a burst of packets from the TCP sender, which may result in multiple packet losses, thus delaying the service response time. To address the ACK compression problem, several methods have been proposed, such as ACK congestion control [53], ACK filtering [54], and ACK prioritization [55]. Regarding mobile networks, Chan and Ramjee [56] introduced an ACK regulator to alleviate the impact of ACK compression. The regulation algorithm assures that the TCP sender is operated mainly in the congestion avoidance phase by limiting the maximum number of buffer overflow losses to one. In the algorithm, the ACK regulator sends an arriving ACK back toward the TCP sender only when the available buffer space is enough at least for one data packet. Therefore, the ACK regulator allows at most one buffer overflow packet loss. The ACK regulator can significantly improve the TCP throughput while slightly increases the RTT. As a result, it reduces the overall service response time.

However, the methods based on ACK management may reduce the performance of TCP sender due to the delayed ACK. To minimize the impact of ACK regulations on the TCP sender, Ming-Chit *et al.* [57] proposed ACE (Acknowledgement based on CWND Estimation), which varies the number of acknowledged packets per ACK according to the sender's CWND. The larger the CWND, the more acknowledged packets per ACK, and vice versa. Therefore, the number of ACK sent on the uplink is reduced without introducing much impact on the TCP sender.

5) Session Layer Techniques: The goals of session layer optimization are three-folds: to reduce the number of DNS lookups, to optimize the number of concurrent TCP connections, and to minimize the TCP idle time.

Reducing DNS lookups: Web based Internet applications usually access content from different servers. To download the content, the DNS protocol is frequently used to resolve the server names. Considering the large RTT of wireless networks, frequent DNS queries introduce significant delays to web service. Therefore, reducing the number of DNS lookups can accelerate the web based applications. One method is to configure the web browser to explicitly point to a cache proxy. In this way, the mobile user only need to lookup the DNS of the cache proxy, and only open connections with the cache proxy. The cache proxy will perform all the DNS queries over wireline networks, thus reducing the service response latency. Another method is to rewrite the URLs contained in the HTML [58]. In this method, the rewriting proxy intercepts the HTML file from the web server, and adds the IP address of the proxy server as a prefix to the URLs. As a result, the web clients recognize the proxy server as the only web content server, and thus frequent DNS queries are avoided.

Reducing concurrent TCP links: Concurrent TCP connections may deteriorate the web performance in wireless networks. Session layer techniques include URL rewriting, DNS rewriting, and content bundling. As discussed in the previous paragraph, after URL rewriting, the web clients only open TCP connections with the proxy server, thus minimizing the number of TCP connections. In DNS rewriting, the DNS rewriting proxy intercepts the DNS servers' responses and attach the IP address of the cache proxy at the top of the returned IP lists [58]. As a result, the cache proxy is the first choice for the web client. The IP addresses of the original servers work as backups in case the cache proxy is unavailable due to mobility. Therefore, the number of TCP connections can be reduced in the same way as explained in the URL rewriting. Content bundling groups all the embeded objects in a single file. Therefore, the web clients do not have to establish multiple TCP connections to download the embeded objects from different servers.

Reducing TCP idle time: With limited processing abilities, the mobile devices may require relevant long time to parse the received data, and then send the next request. As a result, the TCP sender may stay in an idle status waiting for the incoming requests. Waiting in the TCP idle status delays the response time of web based applications. To address this problem, Gomez *et al.* [59] introduced the parse and push proxy that parses the HTML for the mobile users and pushes the enlisted objects toward them prior to the requests. On the client side, the mobile users always check their local cache before requesting the content; if the contents have been pushed into their local cache by the proxy, the mobile users retrieve the content from the cache; otherwise, the mobile users send the requests to the server. This method reduces both the TCP idle time and the number of RTTs required to download the web pages.

C. Cross Domain Techniques

Cross domain techniques optimize the interactions between applications/content and the underlying networks. These techniques can be classified into two categories: content adaptation techniques and protocol adaptation techniques.

1) Content Adaptation Techniques: Content adaptation techniques accelerate the mobile web services by tailoring the original content to fit the mobile networks and mobile devices. The benefits of content adaptations are two folds: for the network, the content adaptations help to reduce the data volume of web pages, thus alleviating the long latency

caused by the low bandwidth; for the mobile device, by fitting the content to mobile devices' capabilities such as screen size, memory size and processing speed, the content adaptations save the time spent on rendering the complex web pages on mobile devices [60]. The content adaptation techniques can be classified into three categories: proxy-based content adaptation, application-aware adaptation, and page-layout adaptation [61].

- **Proxy based adaptation:** this approach does not require any modification on either the content server or the web browsers. All the web communication are redirected to the proxy, where the content are adapted according to the the characteristics of mobile devices and the network conditions. The proxy decides the appropriate adaptations to mobile devices [62].
- **Application-aware adaptation:** in application-aware adaptation, applications and the operating system cooperate to adapt the content for subscribers. The operating systems carry the centralized content adaptations at the system level such as monitoring the resource level and arbitrating the application concurrency, while individual applications take care of application specific adaptation such as selecting the fidelity of the contents [63].
- **Web page layout adaptation:** in web page layout adaptation, the layouts of the web are reconstructed to fit the users' preferences. Web page layout adaptations can be implemented either in the proxy [64] or in the mobile devices [65]. If the adaptations are implemented in the proxy, the proxy will re-author the web content according to the predefined rules and policies. If the adaptations are installed on mobile devices, they provide subscribers a summary of the web content, from which the subscribers can choose the extended views of the web page.

2) Protocol Adaptation Techniques: The interactions between the application behavior and the underlying protocols have a significant impact on network performance, and if not properly considered, they may completely negate the improvement achieved by optimizing the underlying protocols [66]. Therefore, application awareness is required to accelerate the delivery in wireless networks. Application aware acceleration techniques can recognize the application types, and optimize the underlying protocol according to the characteristics of the applications. A^3 [66] is an application aware acceleration software that can recognize FTP, CIFS and SMTP and HTTP protocol based on the session thin messages, and optimize them accordingly. A^3 integrates five techniques: transaction prediction and prefetching to predict the users' next requests and pre-retrieve them, redundant and aggressive retransmission to protect session control messages from loss, prioritized fetching to fetch important data first with respect to the application performance, infinite buffering to prevent a network connection termination because of the small receiving buffer of mobile devices, and application aware encoding to use application specific information to better encode or compress the data. Upon recognizing the application, these techniques are specified according to the application's characteristics, thus achieving better performance than application-blind acceleration.

V. Mobile Data Offloading

Mobile data services usually experience longer latency because mobile networks have lower bandwidth and less reliable connections than wireline networks. Mobile data offloading enables subscribers to utilize high rate networks, such as WiFi, to retrieve the content, thus accelerating content delivery. There are two mechanisms that can be applied to offload mobile data traffic: direct data offloading and network aggregation.

A. Direct Data Offloading

Existing mobile data traffic offloading mechanisms can be classified into two categories. The first category of mechanisms is offloading mobile traffic through opportunistic communications. In this method, instead of downloading content from infrastructure, mobile users can share their content with each other via peer-to-peer collaborations. The other explores the metro scale WiFi networks to offload mobile data traffic.

1) Offloading through opportunistic communications: This type of traffic offloading mechanisms relies on the mobile users to disseminate the content. Given a set of mobile users who request some content from the servers, the offloading mechanism selects a subset of users and deliver the content to them through cellular networks. The subset of users further disseminates the content through opportunistic ad-hoc communications to those users outside the subset. The key designing issue is to select the subset of users in order to maximize traffic offloading. Han *et al.* [67] proposed a mechanism to select the subset of users based on either users' activities or mobilities. However, their mechanism does not guarantee the performance of content delivery in term of delay. To meet the delay requirements, Whitbeck *et al.* [68] proposed to push the content to the users through cellular networks when ad-hoc communications will fail to deliver the content within some target delay. Li *et at.* [69] analyzed the mobile data offloading problem mathematically. They formulated the offloading problem as a submodular function maximization problem, and proposed several algorithms to achieve the optimal solution. To encourage mobile users participate in the traffic offloading, Zhou *et al.* [70] proposed an incentive framework that motivate users to leverage their delay tolerance for cellular data offloading. Mashhadi *et al.* [71] proposed a proactive caching mechanism for mobile users in order to offload the mobile traffic. When the local storage does not have the requested content, the proactive caching mechanism will set a target delay for this request, and explores opportunities to retrieve data from the neighboring mobile nodes. The proactive cache mechanism requests data from cellular networks when the target delay is violated.

2) Offloading through metro scale WiFi: The other type of mechanisms is to explore the metro scale WiFi networks to offload cellular data traffic. Lee *et al.* [72] showed from their measurement results that a user is in WiFi coverage for 70% of the time on average, and if users can tolerate a two hour delay in data transfer, the network can offload about 70% cellular traffic to WiFi networks. Deshpande *et al.* [73] compared the performance of 3G and metro scale WiFi for vehicular network access and showed that even though suffering

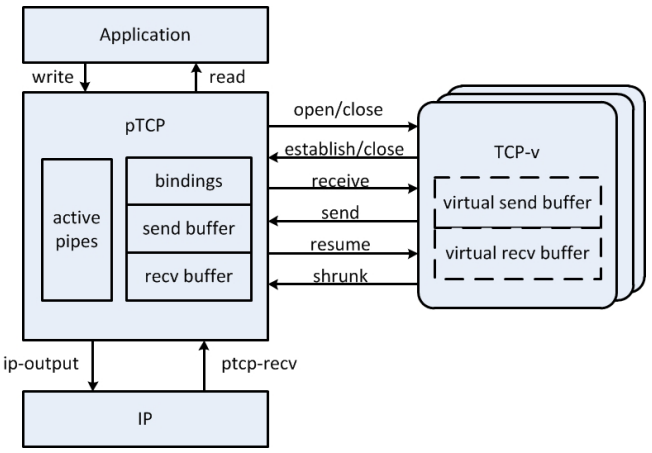


Fig. 10. pTCP structure [78].

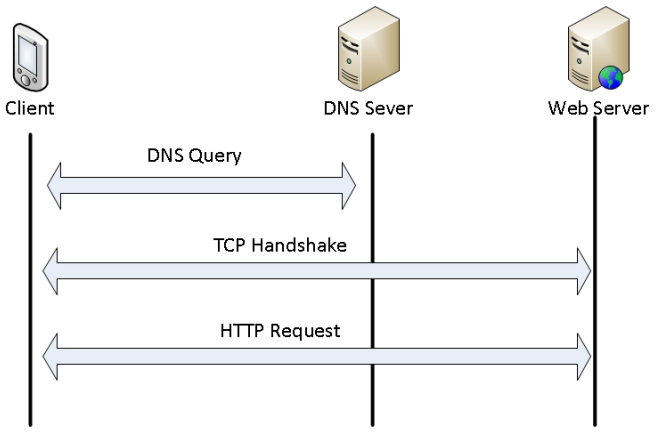


Fig. 11. Web connection procedure.

frequent disconnections, WiFi delivers high throughput when connected. Gass *et al.* [74] pointed out that the full potential of WiFi access points in term of the transmission rate is not reached and is limited to the rate of the backhaul connection to which the access point is connected. To eliminate backhaul bottlenecks, Gass *et al.* [75] proposed In-Motion proxy and Data Rendezvous protocol to enable download large amounts of data during short period opportunistic WiFi connections. In-Motion proxy prefetches the users' requesting data from the original server to its local cache, and streams the data to the users when they are connected to WiFi access points. In-Motion proxy enables large data transfers to be completed with several short connection durations. Data Rendezvous protocol eliminates the bottlenecks between In-Motion proxy and WiFi access points through access point prediction and selection. Balasubramanian *et al.* [76] proposed to offload the delay tolerance traffic such as email and file transfer to WiFi networks. When WiFi networks are not available or experiencing blackouts, data traffic is fast switched back to 3G networks to avoid violating the applications' tolerance threshold. Han and Ansari [77] designed a content pushing system which pushes the content to mobile users through opportunistic WiFi connections. The system responds a user's pending requests or predicted users' future requests, codes these requested contents by using Fountain codes, predicts the user's routes, and prelocates the coded content to the WiFi access points along the user's routes. When the users connect to these WiFi access points, the requested content is delivered to the users via the WiFi connections.

B. Network Aggregation

Network Aggregation techniques enable users to utilize multiple radio interfaces to download content and thus reducing the content delivery delay. In this subsection, we describe recent network aggregation methods: pTCP [78], MAR [79], and Supper-aggregation [80].

1) pTCP: pTCP is a parallel TCP structure that aggregates bandwidth of different networks. As shown in Fig. 10, pTCP structure contains pTCP, which acts as a central engine that interacts with applications, IP networks, and TCP-v. pTCP creates one TCP-v as a virtual TCP pipe for each radio interface. TCP-v relies on traditional TCP protocol to realize reliable transmission. pTCP maintains the central send and receive buffers across all the pipelines, and schedule the transmission over all the TCP-v pipelines according to their link conditions.

2) MAR: MAR is a mobile access router that provides internet access to users on a commuter bus via aggregating the bandwidth of multiple WWAN networks. Mobile users use short range wireless interfaces such as WiFi and Bluetooth to connect to MAR. MAR routes users' requests through the aggregated radio links. The advantage of MAR is that it does not require any modification on neither users' devices nor content servers.

3) Super-aggregation: Super-aggregation aims intelligently to aggregate multiple radio interfaces to achieve better network enhancements. Instead of simply aggregating the radio interfaces, super-aggregation assigns different radio interfaces with different tasks based on their link characteristics. It contains three principles: selective offloading, proxying, and mirroring. With respect to selective offloading, super-aggregation offloads the ACK messages of WiFi networks to 3G networks to addresses the ACK congestion in the uplink of WiFi networks. It utilizes the 3G link as a monitoring proxy that reports the blackout events to the WiFi networks. Upon receiving the report, WiFi link freezes the current TCP status to avoid CWND reductions. Regarding mirroring, super-aggregation utilizes 3G links as a redundant link to fetch the loss packets.

VI. Web Content Delivery Acceleration System

The procedure of provisioning wireless web services is shown in Fig. 11. Before the web clients can fetch data from the server, they have to conduct DNS query and TCP handshake processes, which costs at leat two RTTs. Given the large RTT of the mobile networks, such processes introduce significant delays to the service response time. In addition, the clients are prone to open multiple TCP connection simultaneously, which results in the RTT inflation as discussed in the previous section. To accelerate the web content delivery, we have to reduce the number of RTTs required to download the web pages, and to shorten the length of RTTs. In this section, we overview the existing web content delivery acceleration systems.

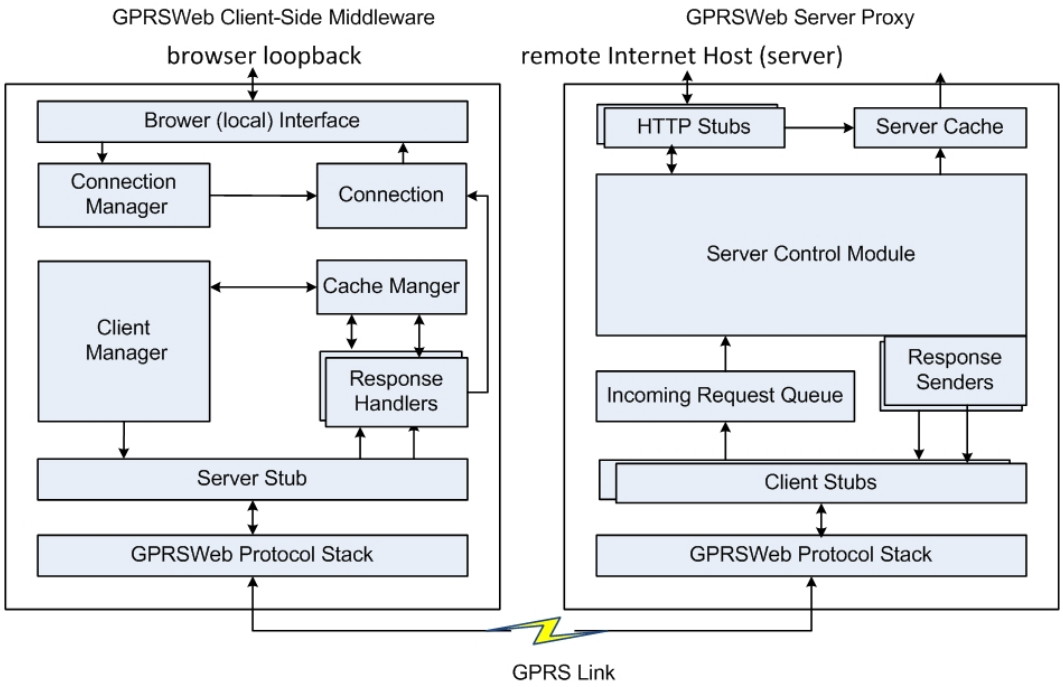


Fig. 12. GPRSWeb [28].

A. Web Acceleration System

Web based applications are among most popular mobile applications. Therefore, accelerating web content delivery is essential in mobile networks. Web acceleration systems that integrate several optimization techniques at different layers in OSI stack are effective tools for the web delivery acceleration. In this subsection, we discuss and compare several of the existing web acceleration systems in mobile networks.

1) GPRSWeb: GPRSWeb [28] is a proxy based web optimization system. As shown in Fig. 12, GPRSWeb applies a dual proxy architecture consisting of a link-aware middleware in mobile devices and a server proxy located in the wired-wireless border. At the client side, the Connection Manager takes users' requests from the web browser and passes them to the Connection module. This module checks whether the requested content is cached by the Cache Manager. If there is no matched content, the Connection module invokes a Server Stub to issue a request to the server proxy, and a Response Handler to process the reply. The response handler also interacts with the cache manager to update the cache state. In the server proxy, Client Stubs receive the clients' requests and pass them to the Server Control Module. This module checks the Server Cache for responses to clients' requests. If the objects are not cached in the server cache, the Server controller invokes HTTP Stubs to download the objects from the remote Internet server. Then, these objects are sent to the Client Stub through the Response Sender. The client stubs coordinate data compression and other optimizations before the response is finally sent back to the client proxy. GPRSWeb has four main mechanisms to improve the web performance over GPRS networks. First, GPRSWeb applies the UDP-based transport layer Protocol to accelerate the web content delivery. The GPRSWeb transport protocol runs between the server proxy the the client side middleware. By incorporating the UDP based protocol, GPRSWeb mitigates the problems of TCP, such as slow startup and ACK compression, and achieves better bandwidth utilization. By relying on RLC layer retransmission, the UDP based transport protocol realizes reliable message transfer. Second, the extended caching is integrated to reduce the number of RTTs required to download the web pages. The extended caching is a SHA (Secure Hash Algorithm) based caching protocol that can effectively increase the hit rates of the client caching. Third, GPRSWeb applies data compression techniques to reduce the traffic volume over wireless networks. It utilizes gzip to compress the raw data while using delta encoding algorithms to update the cached contents. Fourth, the parse-and-push mechanism is implemented to reduce the idle period of the wireless link during a web page download. The proxy server parses the HTML files and pushes the contents into the clients' local cache before they are requested.

2) WebAccel: WebAccel [81] is a client side web service enhancement system that integrates three optimization techniques: the prioritized fetching, object reordering, and connection management. WebAccel consists of three function modules: prioritized fetching, object reordering, and connection management. The prioritized fetching mechanism fetches objects according to their priority levels. In WebAccel, on-screen objects are granted higher priority than the off-screen ones. Therefore, the on-screen objects are always fetched with higher priorities. As illustrated in Fig. 13, the on screen objects−objects1, 2, and 3−are fetched first. In this way, WebAccel addresses the screen contention problem, and thus improves the user-perceived response time. The object reordering algorithm improves the bandwidth utilization through the inter connections reordering and the intra connection reordering.

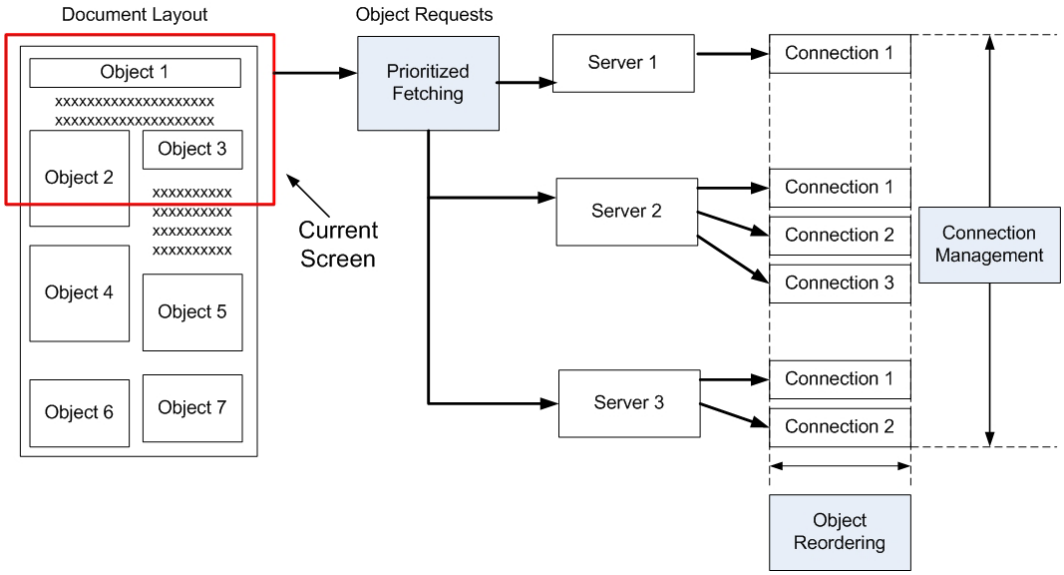


Fig. 13. WebAccel [81].

Inter connection reordering reschedules the object requests across the connections to balance the load distribution over concurrent TCP connections. While the intra connection reordering reorders the sequence of the object requests according to the TCP status. For example, WebAccel sends small objects in the slow start phase of TCP and sends the larger objects in the congestion avoidance phase. The connection management mechanism maintains an optimal number of concurrent TCP connections to improve the bandwidth utilization as well as to mitigate the service delay caused by self-congestion. As shown in Fig. 13, the connection management mechanism maintains three TCP connections to optimize the bandwidth utilization.

3) Nett-Gain: The Nett-Gain platform is a commercialized mobile network acceleration system which provides optimization at both the transport layer and application layer [82]. The system protocol stack is shown in Fig. 14. It supports both a dual proxy model and a clientless model which does not require the middle-ware at the client side. At the application layer, Nett-Gain enhances the transmission rate controller by estimating the network availability using the packet delay and delay derivatives as indicators. Such indicators can reflect the network availability timely and quantitatively, and thus enable an accurate insight of the network congestion. The dual proxy model adopts WBST (Wireless Boosted Session Transport) [83], which is a UDP/IP based transport protocol. WBST does not requires three way handshake and eliminates TCP slow start. For the clientless mode, TCP+ is applied to maintain the semantics and form of the standard TCP with the improved rate controller. At the application layer, the Nett-Gain platform optimizes the HTTP protocol by incorporating GetAll for the dual proxy mode and HTTP+ for the clientless mode. However, details of the techniques are not disclosed to the public. In addition, Nett-Gain applies various compression techniques, such as image reduction, animation reduction, header and request compression, and so on.

4) Macara: The Bytemobile Macara platform [59] is another commercialized web content delivery accelerator which accelerates the mobile networks by reducing the number of RTTs for data transfer in the transactions with TCP and HTTP. The HTTP protocol is replaced by the Macara Dynamic Interleaving technology that minimizes the latency of web page downloading. The platform imposes the use of multiple TCP connections and merges multiple data requests and responses. Extended caching and content format adaption is also applied.

5) NPS: NPS (Non-interfering Prefetching System) [84] is a prefetching system for web service that can avoid the interferences between prefetch and demand requests at the server as well as in the networks. NPS only uses the spare resources to prefetch the web contents. To avoid interferences with demand request at the server level, NPS monitors and restricts the prefetch load on the server. To avoid the interferences at the network level, it applies TCP-Nice [85] which is a congestion control protocol designed for low priority network traffic. NPS gives lower priority to the prefetching requests than the demand requests to avoid delay incurred by the prefetching requests in rendering of demand pages. Aggressive prefetching may result in the cache pollution, and thus reduces the cache efficiency. To address this problem, NPS applies two heuristics to regulate the prefetching. First, it limits the ratio of prefetched bytes to demand bytes sent to a client. Second, it sets the *Expires HTTP* header to a value in the relatively near future to encourage clients to evict the prefectched content earlier.

6) pTHINC: pTHINC [86] is a PDA thin client system that leverages the more powerful web server to render the web pages, and then send the simple screen updates to the PDA for display. It consists of a client reviewer and a pTHINC server. The client reviewer takes the input commands from the users, and send them to the pTHINC server. The server processes the command, and virtualizes and resizes the display, and then sends the screen updates back to the clients. pTHINC provides the user a persistent web session that allows the user to reconnect to a web session after a disconnection. This functions benefits the mobile users who experience

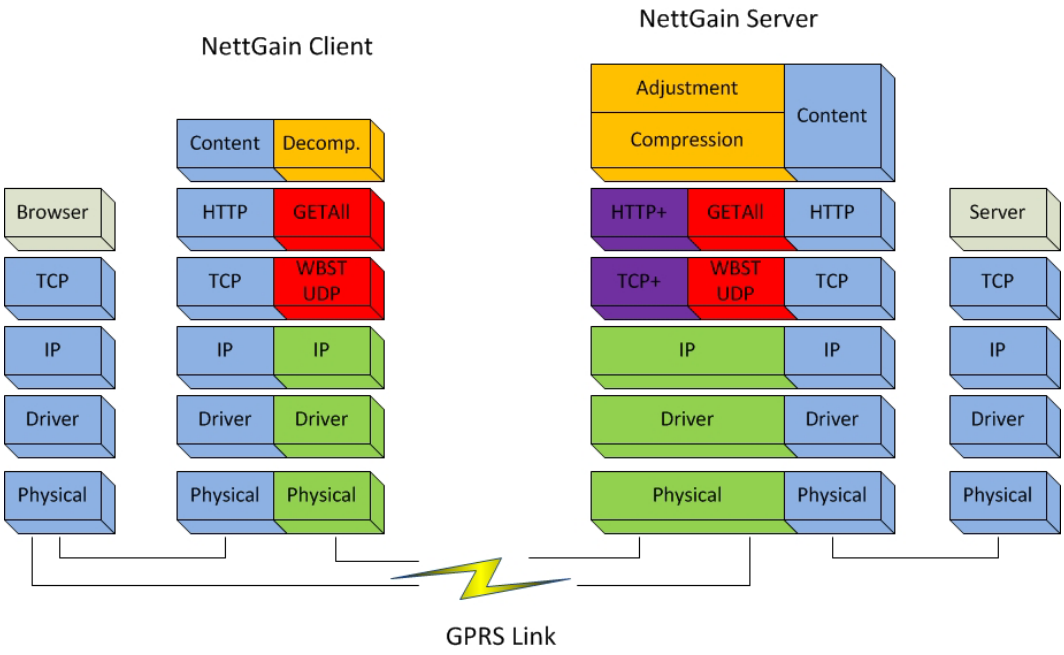


Fig. 14. Nett-Gain platform [82].

intermittent wireless connections. pTHINC can achieve up to 8 times better performance than the native browser in terms of the web service latency.

7) Chameleon: Chameleon [87] is a URICA (Usage-awaRe Interactive Content Adaptation) prototype that adapts image-rich web pages for browsing over bandwidth limited wireless links. It trades the image fidelity for the image loading latency. URICA consists of three components: the client application, the adaptation proxy, and the content server. The adaptation proxy adapts the images according to the predictions from the prediction engine that uses the statistics-based policies and the personalized adaptation schedule. The client application allows users to send feedback to the adaptation proxy if they are not satisfied with the image quality. By receiving the feedback, the adaptation server enhances the image fidelity accordingly and saves the user's preferences for future references. Chameleon can save up to 65% latency and up to 80% bandwidth consumptions for mobile web browsing.

8) Silo: Silo [88] is designed to minimize the number of HTTP requests needed to render a web page. Silo leverages the Javascript and DOM (Document Object Model) storage to minimize the number of HTTP requests. Silo applies the DOM storage to allow the web page maintain a key value database on the client. As illustrated in Fig. 15b, when a web client requests a silo enabled web page, the server responds with a small Javascript that checks the available chunks on the client side and sends back the information to the server. Then, the server sends a lists of the chunks contained in the web page as well as the missing chunks to the client. As compared to the standard HTTP protocol shown in Fig. 15a, Silo reduces the number of RTTs required to fetch the objects contained in the web page by minimizing the HTTP requests, and avoids TCP slow start for unnecessary HTTP connections.

B. Web Acceleration System Summary

The optimization techniques that are adopted in the web acceleration systems are summarized in Table I.

VII. Multimedia Content Delivery Acceleration

The media streaming protocols can be classified into two catalogs: push-based protocols and pull-based protocols [89]. The push-based protocols are UDP-based, e.g., RTSP, in which the media server continually streams the packets to the client until the session is torn down. The pull-based protocols are TCP-based, e.g., HTTP, in which the client requests the content actively from the media server. The HTTP streaming that relies on TCP is preferred for internet video applications due to its lower cost and easier implementation [90]. Popular video sharing websites, e.g. YouTube, YouKu, and MySpace, almost exclusively apply HTTP streaming . The leading solution providers, e.g. Microsoft [90], Apple [91], and Adobe [92], have proposed their video delivery applications based on HTTP. The standardization organizations IETF and 3GPP also proposed specifications on HTTP based streaming [93], [94]. In this section, we overview the techniques that accelerate HTTP based multimedia content delivery.

A. Adaptive Streaming

Two approaches have been proposed for HTTP-based video streaming: progressive download and adaptive streaming. The progressive download approach simply transfers the entire video file as soon as possible. The client requests the video content with certain bit rates according to the client's available bandwidth. The server responds with the requested video content. When the client's playback buffer is filled, the client starts playing the media. At the same time, it continues downloading the media in the background with the same bit rates until it finishes the download or the user tears down

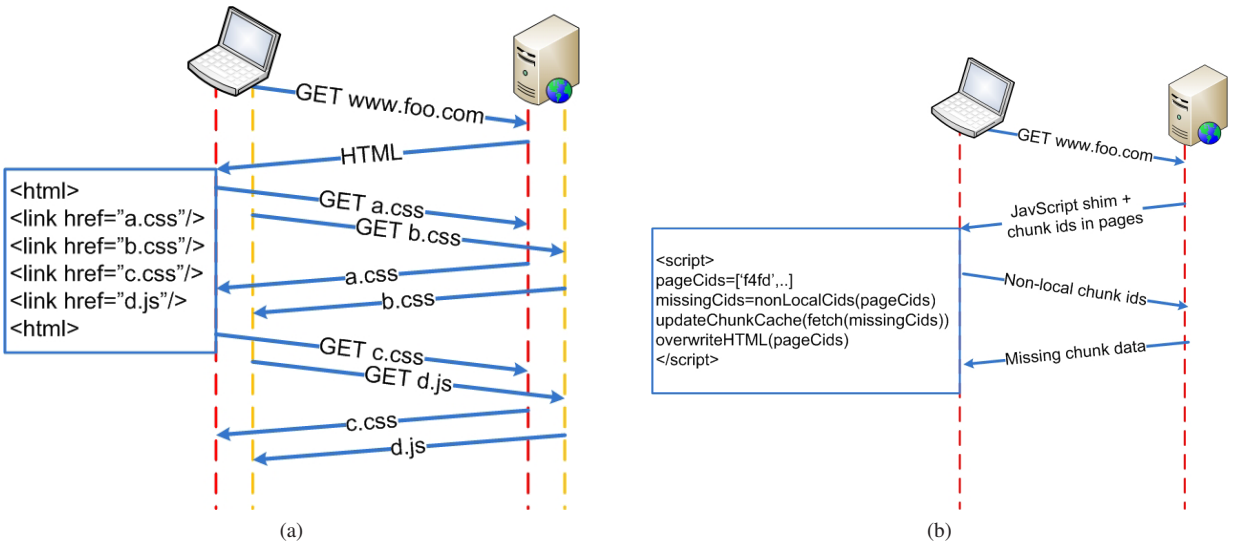


Fig. 15. Silo protocol [88]: (a) the standard HTTP protocol; (b) the Silo protocol.

TABLE I
OPTIMIZATION TECHNIQUES ADOPTED IN VARIOUS WEB ACCELERATION SYSTEMS

Systems	Proxy Module	Acceleration Techniques
GPRSWeb [28]	Dual Proxies	UDP-based transport protocol, SHA-based caching, parse and push, data compression, delta encoding.
WebAccel [81]	Client side proxy	Object reordering, TCP optimization, prioritized fetching, inter connections load balancing.
Nett-Gain [82]	Dual or single Proxies	TCP and HTTP optimization, image reduction, animation reduction, request compression.
Macara [59]	Dual Proxies	TCP and HTTP optimization, caching, content adaption.
NPS [84]	Single proxy	TCP optimization, interference-aware prefetching.
pTHINC [86]	Dual proxies	Thin client protocol, content adaptation.
Chameleon [87]	Dual proxies	Image quality feedback protocol, content adaptation.
Silo [88]	Client Side Proxy	Silo application layer protocol, DOM storage caching.

the connection. The client can play the media smoothly as long as the playback rate is not larger than the download rate. However, to ensure good performance, streaming over TCP requires the network bandwidth that is twice that of the video rate [95]. Such bandwidth requirements are unable to be guaranteed, especially in mobile networks. As a result, if the available bandwidth shrinks, the client may suffer a long latency or frequent interruption during the playback. Addressing the problem with progressive download, adaptive streaming allows either the client or server to adjust the media bit rates according to the available bandwidth, and can alleviate the performance deterioration caused by bandwidth variations. Adaptive streaming can be achieved through stream switching mechanisms, in which the video content is encoded into different bit rates and quality levels, and is partitioned into fragments of a few second long. The outgoing video segments are switched among alternate encodings of a stream based on the either the server's estimations or the client's requests.

Adaptive streaming is an essential video content delivery techniques in mobile networks where users usually experience unstable wireless channel with bursty packet losses that result in various available bandwidth for the video streaming applications. To accommodate the adverse channel condition, adaptive streaming is applied to switch to a lower bit rate streaming, thus reducing the congestion and sustaining the video streaming [96]. Therefore, video adaptive mechanisms are the key component of HTTP-based adaptive streaming. They are expected to timely and accurately map the channel conditions to the bit rates of the video files. Otherwise, the large reaction delay of the rate-based adaptive mechanism can either degrade the user perceived quality, or introduce a delay spike and freeze the streaming. The adaptive mechanisms can either be implemented at the client side or at the server side. We discuss these mechanisms respectively.

1) Client Side Mechanisms: Client side adaptive algorithms are usually implemented in the video player. The sever informs the clients about the available audio and video bit rates and resolutions through the manifest file. The player requests the media content with proper bitrates according to the estimated available bandwidth. We discuss the adaptive algorithms of two commercialized video player: Microsoft Silverlight and Netflix Player.

Microsoft Silverlight smooth streaming player starts with requesting lowest bitrates, and gradually increases the requesting bitrates to the highest available bitrates. The smooth streaming player uses several per-fragment throughput to estimate the available bandwidth rather than using the latest fragment throughput. Such rate adaption mechanism introduces a reaction delay. As a result, Microsoft Silverlight tends to neglect the short bandwidth spikes, e.g., 2 seconds spike

and 5 seconds spikes, and reacts to relevant large spikes (10 seconds) till 40 seconds after the spikes [97].

As compared to Microsoft Silverlight smooth streaming player, Netflix player is more aggressive [97]. It estimates the negative bandwidth spikes by using a smoothed version of the underlying per-fragment throughput while estimating the positive bandwidth spikes based on the instantaneous measurements. Therefore, Netflix player attempts to deliver highest possible encoding rate even when the bandwidth is not sufficient, and it neglects the negative bandwidth spikes. To compensate for the lack of the available bandwidth, Netflix uses a large playback buffer size, which results in a long start up latency.

Although both Microsoft Silverlight smooth streaming player and Netflix player fail to adjust their bitrates to the available bandwidth timely, they perform very well in wired networks. Because wired networks usually have large bandwidth, the playback buffer can be filled quickly after the short lived negative bandwidth spikes. However, the wireless networks, which have only limited bandwidth, may require relative long time to fill the playback buffer. As a result, the streaming can be freezed while the playback buffer drains away.

One reason for the large reaction delay is that the video players tend to estimate the available bandwidth based on a smoothed version of the underlying throughput rather than the latest throughput. These estimators work well in wired networks, but they may cause problems in wireless networks as discussed above. In addition to available bandwidth estimators, the protocol latency delays the application layer actions in adapting to the available bandwidth. To improve the bandwidth utilization, the servers usually maintain a large send buffer. The data in the send buffer may wait for several RTTs until they can be sent out, and thus the data formats may not reflect the current network conditions. Tuning the send buffer can significantly reduce the TCP protocol latency [98] , and therefore may reduce the reaction delay.

2) Server Side Algorithms: There are three major categories of server side adaptive algorithms: application-layer bandwidth estimation based algorithms, TCP stack-based bandwidth estimation based algorithms, and priority streaming algorithms [99].

- **Application layer bandwidth estimation**: this approach estimates the available bandwidth based on the time spent to transmit a specific media content in the blocking TCP socket [100]. As shown in Fig. 16, the delivery process sends the media content to a blocking TCP socket, and writes the delivery rate statistics to share memory segments periodically. The application process reads the statistics from the shared memory segments, and configures the media transcoder accordingly.
- **TCP based bandwidth estimation**: this method utilizes TCP statistics such as current CWND, RTT and ACKs to estimate the available bandwidth [101]. Argyriou [102] proposed a simple model that estimates the available bandwidth and expected content deterioration based on TCP statistics, and then encodes the media content accordingly.
- **Priority Streaming**: priority streaming [103] is designed to improve the quality of video in a period of time, which is guarded by the so called deadline. The video stream is split into segments. The video syntax elements such as slices, frames, and layers are reordered according to their priorities. Each segment has a sending deadline, after which, the server discards the current segment and switches to the next segment. The video quality depends on how many video syntax elements are received for the video segment before its sending deadline.

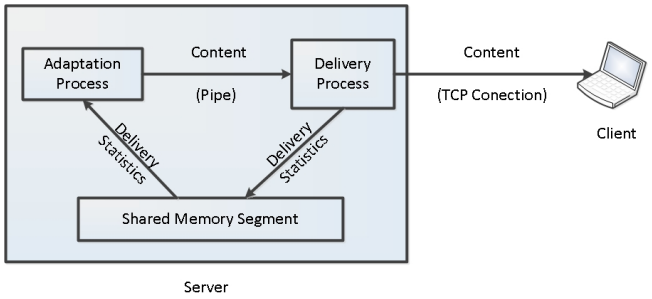


Fig. 16. Application layer bandwidth estimation [100].

B. Other Methods

Besides adaptive streaming, many other techniques can be applied to enhance the multimedia delivery over mobile networks. In this subsection, we discuss several popular techniques that can effectively improve the delivery latency.

1) Multicast Scheduling Optimization: Multicast is an efficient method to deliver multimedia content to a group of users who request the same content. Instead of scheduling an individual user at each time slot, multicast enables NodeBs to schedule a group of users at a time slot. Therefore, mutlicast not only minimizes the utilization of network resources, but also reduces the delivery latency of mutlimedia content. However, in multicast, NodeBs can only transmit to one multicast group at one data rate. Because subscribers in the multicast group may experience different channel fading, the users' achievable data rates are different. NodeBs have to transmit at the lowest achievable data rates of the users in the group; otherwise, the users with the largest channel fading cannot decode the information. This multicast scheme limits the throughput of users with small channel fading, and thus may degrade their QoE. Won *et al.* [104] proposed a multicast proportional fair scheduler to improve the multicast throughput. The idea of the multicast scheduler is to let the users with poor channel quality receive a base level of service and enhance the service quality of the users with good channel. The multicast proportional fair scheduler improves the multicast throughput and accelerates the content distribution in mobile networks.

2) Multiple HTTP/TCP Connections: in this approach, the client opens multiple HTTP/TCP connections with the server. One of the advantages of this method is that multiple connections can avoid the delay caused by TCP slow start. The other advantage is that multiple connections can alleviate the throughput reduction effect caused by packet loss since the probabilities that all concurrent TCP connections experience packet loss are small. Therefore, Multiple HTTP/TCP connections can alleviate the streaming performance deterioration

on Internet caused by packets loss [105], [106], and by insufficient bandwidth [107]. However, in wireless networks, the concurrent TCP connection may result in self-congestion, and thus introduce additional latency.

3) Priority-based Protection: in this method, the media contents are given different protection in terms of FEC and retransmission opportunities according to their importance. In [108], frame priorities are associated with the RLC layer protection in 3G networks. Important frames are given higher priorities, and thus gain more protection in term of more retransmission opportunities. In [109], different layers of the layered video content are granted different error protections by limiting the retransmission attempts according to their priorities.

4) Multiple Path Aggregation: As discussed in the previous section, network aggregation can increase the bandwidth and reliability of wireless networks. Such advantages benefit media delivery in wireless networks. Miu *et al.* [110] proposed to use multiple channel simultaneously or to switch among them based on the channel condition to reduce the media transmission latency over WLAN. In [111], [112], the authors proposed to exploit the diversity of multiple WWAN and to aggregate lower capacity wide area data channels together to create a single high bandwidth channel for multimedia applications. Kaspar et al [113] used HTTP range requests to download video segments over multiple wireless links.

5) Cooperative Delivery: In wireless networks, users may experience bad channel quality due to wireless fading and the distance between the users and the base stations. Such adverse environment causes packet losses and introduces additional delay. Relying nearby wireless peers to relay the packets can be one effective method against the erroneous channel conditions. Li *et al.* [114] proposed a multi source streaming system that leverages the wireless peers by having them form a joint sender group with the server. Lu *et al.* [115] proposed an opportunistic retransmission protocol by using the overhearing nodes as relays to retransmit the failed packets on behalf of the source.

VIII. Conclusion

In this survey, we have presented an overview on content delivery acceleration solutions in mobile networks. By studying the live network measurements, we have identified the major obstacles that delay the content delivery over mobile networks. Afterward, we have classified the solutions on accelerating content delivery in mobile networks into three categories: mobile system evolution, content and network optimization, and mobile data offloading, and discussed content delivery acceleration solutions in each category, respectively. Considering the web service and video streaming as two of the most important services in future mobile networks, we have provided an overview of web content delivery acceleration systems, and discussed the HTTP-based adaptive streaming techniques. Therefore, this survey is a useful reference for further investigation.

In recent years, tremendous efforts have been paid on accelerating service delivery in mobile networks from both academia and industries. However, mobile network latency is still too large to satisfy users' expectations. There are still many open challenges to be answered. For the web service delivery, to our best knowledge, most of the existing web acceleration systems do not integrate the network layer acceleration techniques. Besides, the interactions of different content delivery acceleration approaches are seldom studied. In addition, how to implement the prefecthing mechanism with a reasonable pricing policy is still under investigation.

For the multimedia delivery, the current stream adaptive algorithms are unable to track the time-varying network capacities. The measurement of Akamai HD Streaming services showed that it took about 150 seconds to match the video bitrates to the available bandwidth. Such delay may result in longer playback latency. The client indicates the negative bandwidth spikes to the server with a delay of about seven seconds, and the server takes another seven seconds to respond [116]. Such reaction delay may result in playback interruptions. The adaptive control algorithms developed for UDP based streaming can be borrowed; however, since TCP has its own congestion control mechanism, the correlation between the TCP control and adaptive encoding should be studied carefully. In addition, with the increasing popularity of HD (High Definition) and 3D videos, users expects to view these high quality video readily over wireless networks. Since most of existing video delivery systems trade video quality for delivery latency, ensuring a low latency without sacrificing the video quality seems an impossible mission for currently available systems.

References

[1] J. Skorupa, "Forecast: Application acceleration equipment, worldwide, 2007-2015, 1q11 updated," Mar. 2011. [Online]. Available: http://www.gartner.com/DisplayDocument?id=1577015

[2] Y. Zhang, N. Ansari, M. Wu, and H. Yu, "On wide area network optimization," *IEEE Commun. Surveys Tuts.*, 2011, DOI:10.1109/SURV.2011.092311.00071.

[3] "Cisco visual networking index: Global mobile data traffic forecast update, 20112016," Feb. 2012. [Online]. Available: http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-520862.pdf

[4] Akamai Technologies, Inc., "Akamai reveals 2 seconds as the new threshold of acceptability for ecommerce web page response times," Sep. 2009. [Online]. Available: http://www.akamai.com/html/about/press/releases/2009/press_091409.html

[5] "Erricsson traffic and market report," Jun. 2012. [Online]. Available: http://www.ericsson.com/res/docs/2012/traffic_and_market_report_june_2012.pdf

[6] "Evolved universal terrestrial radio access (E-UTRA) and evolved universal terrestrial radio access network (E-UTRAN)," 3GPP TS 36.300 version 10.5.0 Release 10, 2011.

[7] "Ericsson mobile cloud accelerator," 2011. [Online]. Available: http://www.ericsson.com/ourportfolio/telecom-operators/mobile-cloud-accelerator?nav=marketcategory002

[8] E. Dahlman, S. Parkvall, J. Sköld, and P. Beming, *3G Evolution: HSPA and LTE for Mobile Broadband*. Academic Press, Elsevier, 2008.

[9] J. Cano-Garcia, E. Gonzalez-Parada, and E. Casilari, "Experimental analysis and characterization of packet delay in UMTS networks," in *Next Generation Teletraffic and Wired / Wireless Advanced Networking*. Springer Berlin Heidelberg, 2006, vol. 4003, pp. 396–407.

[10] A. Mutter, M. Necker, and S. Lück, "IP-packet service time distributions in UMTS radio access networks," in *Proc. 10th Open European Summer School and IFIP WG 6.3 Workshop, EUNICE '04*, Tampere, Finland, Jun. 2004.

[11] J. Prokkola, P. H. J. Perälä, M. Hanski, and E. Piri, "3G/HSPA performance in live networks from the end user perspective," in *Proc. 2009 IEEE International Conference on Communications, ICC'09*, Dresden, Germany, Jun. 2009.

[12] X. Liu, A. Sridharan, S. Machiraju, M. Seshadri, and H. Zang, "Experiences in a 3G network: interplay between the wireless channel and applications," in *Proc. 14th ACM International Conference on Mobile Computing and Networking, MobiCom '08*, San Francisco, California, USA, Sep. 2008.

[13] W. L. Tan, F. Lam, and W. C. Lau, "An empirical study on the capacity and performance of 3G networks," *IEEE Trans. Mobile Computing*, vol. 7, no. 6, pp. 737–750, Jun. 2008.

[14] P. Romirer-Maierhofer, F. Ricciato, and A. Coluccia, "Explorative analysis of one-way delays in a mobile 3G network," in *Local and Metropolitan Area Networks, 2008, LANMAN '08. 16th IEEE Workshop on*, Sep. 2008, pp. 73 –78.

[15] Y. Lee, "Measured TCP Performance in CDMA 1x EV-DO Network," in *Proc. 7th International Conference on Passive and Active Measurement, PAM '06*, Adelaide, Australia, Mar. 2006.

[16] K. Mattar, A. Sridharan, H. Zang, I. Matta, and A. Bestavros, "TCP over CDMA2000 networks: A cross-layer measurement study," in *Passive and Active Network Measurement*. Springer Berlin / Heidelberg, 2007, vol. 4427, pp. 94–104.

[17] J. Kilpi and P. Lassila, "Micro- and macroscopic analysis of RTT variability in GPRS and UMTS networks," in *Networking 2006: Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications Systems*. Springer Berlin / Heidelberg, 2006, vol. 3976, pp. 1176–1181.

[18] M. Sagfors, R. Ludwig, M. Meyer, and J. Peisa, "Queue management for TCP traffic over 3G links," in *in Proc. 2003 IEEE Wireless Communications and Networking Conference, WCNC '03*, New Orleans, LA, USA, Mar. 2003.

[19] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *IEEE/ACM Trans. Netw.*, vol. 1, pp. 397–413, Aug. 1993.

[20] K. Pentikousis, M. Palola, M. Jurvansuu, and P. Perala, "Active goodput measurements from a public 3G/UMTS network," *IEEE Commun. Lett.*, vol. 9, no. 9, pp. 802–804, Sep. 2005.

[21] H. Holma and A. Toskala, *WCDMA for UMTS: Radio Access for Third Generation Mobile Communications*. John Wiley & Sons, 2000.

[22] J.-C. Chen and T. Zhang, *IP-Based Next-Generation Wireless Networks: System, Architecture, and Protocols*. John Wiley & Sons, 2004.

[23] H. Nishiyama, N. Ansari, and N. Kato, "Wireless loss-tolerant congestion control protocol based on dynamic AIMD theory," *Wireless Commun.*, vol. 17, no. 2, pp. 7–14, Apr. 2010.

[24] P. Benko, G. Malicsko, and A. Veres, "A large-scale, passive analysis of end-to-end TCP performance over GPRS," in *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, Hong Kong, China, Mar. 2004.

[25] R. Chakravorty and I. Pratt, "Performance issues with general packet radio service," *J. Communications and Networks (JCN)*, vol. 4, pp. 266–281, 2002.

[26] ——, "WWW performance over GPRS," in *Mobile and Wireless Communications Network, 2002. 4th International Workshop on*, Stockholm, Sweden, Sep. 2002.

[27] Y. Tian, K. Xu, and N. Ansari, "TCP in wireless environments: problems and solutions," *IEEE Commun. Mag.* , vol. 43, no. 3, pp. S27 – S32, Mar. 2005.

[28] R. Chakravorty, A. Clark, and I. Pratt, "Optimizing web delivery over wireless links: design, implementation, and experiences," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 2, pp. 402 – 416, Feb. 2005.

[29] J. Huang, Q. Xu, B. Tiwana, Z. M. Mao, M. Zhang, and P. Bahl, "Anatomizing application performance differences on smartphones," in *Proc. 8th International Conference on Mobile Systems, Applications, and Services, MobiSys '10*, San Francisco, CA, USA, Jun. 2010.

[30] K. Jang, M. Han, S. Cho, H.-K. Ryu, J. Lee, Y. Lee, and S. B. Moon, "3G and 3.5G wireless network performance measured from moving cars and high-speed trains," in *Proc. 1st ACM workshop on Mobile Internet through Cellular Networks, MICNET '09*, Beijing, China, Sep. 2009.

[31] C. K. Lau, "Improving mobile IP handover latency on end-to-end TCP in UMTS/WCDMA networks," in *Proc. 2005 ACM Conference on Emerging Network Experiment and Technology, CoNEXT '05*, Toulouse, France, Oct. 2005.

[32] E. Nurvitadhi, B. Lee, C. Yu, and M. Kim, "Adaptive semi-soft handoff for cellular IP networks," *Int. J. Wire. Mob. Comput.*, vol. 2, pp. 109–119, July 2007.

[33] S. Mohan, R. Kapoor, and B. Mohanty, "Latency in HSPA data networks," 2011. [Online]. Available: http://www.qualcomm.com/documents/files/latency-in-hspa-data-networks.pdf

[34] P. Lescuyer and T. Lucidarme, *Evolved Packet System (EPS): The LTE and SAE Evolution of 3G UMTS*. John Wiley & Sons Ltd, 2008.

[35] "Extending a content delivery network (cdn)into a mobile or wireline network," Apr. 2012.

[36] A. Anand, C. Muthukrishnan, A. Akella, and R. Ramjee, "Redundancy in network traffic: findings and implications," in *Proc. 11th International Joint Conference on Measurement and Modeling of Computer Systems, SIGMETRICS '09*, Seattle, WA, USA, Jun. 2009.

[37] C. Lumezanu, K. Guo, N. Spring, and B. Bhattacharjee, "The effect of packet loss on redundancy elimination in cellular wireless networks," in *Proc. 10th Annual Conference on Internet Measurement, IMC '10*, Nov. 2010.

[38] J. T. Yao, J. Domnech, A. Pont-Sanjun, J. Sahuquillo, and J. A. Gil, "Evaluation, analysis and adaptation of web prefetching techniques in current web," in *Web-based Support Systems*. Springer London, 2010, pp. 239–271.

[39] J. Zhu, J. Hong, and J. G. Hughes, "Using markov models for web site link prediction," in *Proc. 13th ACM Conference on Hypertext and Hypermedia, HYPERTEXT '02*, College Park, Maryland, USA, Jun. 2002.

[40] Y.-F. Huang and J.-M. Hsu, "Mining web logs to improve hit ratios of prefetching and caching," *Know.-Based Syst.*, vol. 21, pp. 62–69, February 2008.

[41] B. D. Davison, "Predicting web actions from HTML content," in *Proc. 13th ACM Conference on Hypertext and Hypermedia, HYPERTEXT '02*, College Park, Maryland, USA, Jun. 2002.

[42] S. Khemmarat, R. Zhou, L. Gao, and M. Zink, "Watching user generated videos with prefetching," in *Proc. 2nd Annual ACM Conference on Multimedia Systems, MMSys '11*, San Jose, CA, USA, Feb. 2011.

[43] B. Housel, G. Samaras, and D. Lindquist, "WebExpress: A client/intercept based system for optimizing web browsing in a wireless environment," *Mobile Networks and Applications*, vol. 3, pp. 419–431, 1998.

[44] R. Koodli, "Fast handovers for mobile IPv6," July 2005. [Online]. Available: https://www.ietf.org/rfc/rfc4068.txt

[45] Y. Lang, D. Wübben, A. Dekorsy, V. Braun, and U. Doetsch, "Improved HARQ based on network coding and its application in LTE," in *Proc. IEEE Wireless Communications and Networking Conference (WCNC 2012)*, Paris, France, Apr. 2012.

[46] J. Sundararajan, D. Shah, M. Médard, S. Jakubczak, M. Mitzenmacher, and J. Barros, "Network coding meets TCP: Theory and implementation," *Proc. IEEE*, vol. 99, no. 3, pp. 490 –512, Mar. 2011.

[47] M. Kim, M. Médard, and J. Barros, "Modeling network coded TCP throughput A simple model and its validation," in *Proc. International ICST/ACM Conference on Performance Evaluation Methodologies and Tools (Valuetools)*, ENS, Cachan. France, May 2011.

[48] Y. Zhang and L. Qiu, "Speeding up short data transfers: Theory, architectural support, and simulation results," Cornell University, Ithaca, NY, USA, Tech. Rep., 2000.

[49] T. Goff, J. Moronski, D. Phatak, and V. Gupta, "Freeze-TCP: a true end-to-end TCP enhancement mechanism for mobile environments," in *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 3, Tel-Aviv,Israel, Mar. 2000.

[50] P. Rodriguez and V. Fridman, "Performance of PEPs in cellular wireless networks," in *Web Content Caching and Distribution*. Springer Netherlands, 2004, pp. 19–38.

[51] T. Han, N. Ansari, M. Wu, and H. Yu, "TCP-Mobile Edge: Accelerating content delivery in mobile networks," in *Proc. 2012 IEEE International Conference on Communications, ICC'12*, Ottawa, Canada, June 2012, pp. 6921–6925.

[52] K. Xu, Y. Tian, and N. Ansari, "Improving TCP performance in integrated wireless communications networks," *Comput. Netw.*, vol. 47, no. 2, pp. 219–237, Feb. 2005.

[53] S. Floyd, A. Arcia, D. Ros, and J. Iyengar, "Adding acknowledgement congestion control to TCP," July 2009. [Online]. Available: https://tools.ietf.org/html/draft-floyd-tcpm-ackcc-06

[54] H. Balakrishnan, R. H. Katz, and V. N. Padmanbhan, "The effects of asymmetry on TCP performance," *Mob. Netw. Appl.*, vol. 4, pp. 219–241, Oct. 1999.

[55] L. Kalampoukas, A. Varma, and K. K. Ramakrishnan, "Improving TCP throughput over two-way asymmetric links: analysis and solutions," *SIGMETRICS Perform. Eval. Rev.*, vol. 26, pp. 78–89, June 1998.

[56] M. C. Chan and R. Ramjee, "TCP/IP performance over 3G wireless links with rate and delay variation," in *Proc. 8th annual international conference on Mobile computing and networking, MobiCom '02*, Sep. 2002.

[57] I. T. Ming-Chit, D. Jinsong, and W. Wang, "Improving TCP performance over asymmetric networks," *SIGCOMM Comput. Commun. Rev.*, vol. 30, pp. 45–54, July 2000.

[58] P. Rodriguez, S. Mukherjee, and S. Ramgarajan, "Session level techniques for improving web browsing performance on wireless links," in *Proc. 13th International Conference on World Wide Web, WWW '04*, New York, NY, USA, May 2004.

[59] C. Gomez, M. Catalan, D. Viamonte, J. Paradells, and A. Calveras, "Web browsing optimization over 2.5G and 3G: end-to-end mechanisms vs. usage of performance enhancing proxies," *Wireless Communication and Mobile Computing*, vol. 8, pp. 213–230, Feb. 2008.

[60] Y. Chen, X. Xie, W.-Y. Ma, and H.-J. Zhang, "Adapting web pages for small-screen devices," *IEEE Internet Comput.*, vol. 9, no. 1, pp. 50 – 56, Jan. 2005.

[61] I. Mohomed, "Interactive content adaptation," Ph.D. dissertation, University of Toronto, 2009.

[62] B. Knutsson, H. Lu, J. Mogul, and B. Hopkins, "Architecture and performance of server-directed transcoding," *ACM Trans. Internet Technol.*, vol. 3, pp. 392–424, Nov. 2003.

[63] B. Noble and M. Satyanarayanan, "Experience with adaptive mobile applications in odyssey," *Mob. Netw. Appl.*, vol. 4, pp. 245–254, December 1999.

[64] A. Moshchuk, S. D. Gribble, and H. M. Levy, "Flashproxy: transparently enabling rich web content via remote execution," in *Proc. 6th International Conference on Mobile Systems, Applications, and Services, MobiSys '08*, Breckenridge, CO, USA, Jun. 2008.

[65] O. Buyukkokten, H. Garcia-Molina, and A. Paepcke, "Accordion summarization for end-game browsing on PDAs and cellular phones," in *Proc. SIGCHI Conference on Human Factors in Computing Systems, CHI '01*, Seattle, Washington, United States, Apr. 2001.

[66] Z. Zhuang, T.-Y. Chang, R. Sivakumar, and A. Velayutham, "Application-aware acceleration for wireless data networks: Design elements and prototype implementation," *IEEE Trans. Mobile Computing*, vol. 8, pp. 1280–1295, 2009.

[67] B. Han, P. Hui, V. A. Kumar, M. V. Marathe, G. Pei, and A. Srinivasan, "Cellular traffic offloading through opportunistic communications: a case study," in *Proc. 5th ACM Workshop on Challenged Networks, CHANTS' 10*, Sep. 2010.

[68] J. Whitbeck, M. Amorim, Y. Lopez, J. Leguay, and V. Conan, "Relieving the wireless infrastructure: When opportunistic networks meet guaranteed delays," in *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2011 IEEE International Symposium on a*, Lucca, Italy, Jun. 2011.

[69] Y. Li, G. Su, P. Hui, D. Jin, L. Su, and L. Zeng, "Multiple mobile data offloading through delay tolerant networks," in *Proc. 6th ACM Workshop on Challenged Networks, CHANTS' 11*, Sep. 2011.

[70] X. Zhuo, W. Gao, G. Cao, and Y. Dai, "Win-coupon: An incentive framework for 3G traffic offloading," in *Network Protocols (ICNP), 2011 19th IEEE International Conference on*, Vancouver, BC Canada, Oct. 2011.

[71] A. Mashhadi and P. Hui, "Proactive caching for hybrid urban mobile networks," 2010. [Online]. Available: http://www.cs.ucl.ac.uk/research/researchnotes/documents/RN_10_05_000.pdf

[72] K. Lee, I. Rhee, J. Lee, Y. Yi, and S. Chong, "Mobile data offloading: how much can WiFi deliver?" *SIGCOMM Comput. Commun. Rev.*, vol. 41, no. 4, pp. 425–426, Aug. 2010.

[73] P. Deshpande, X. Hou, and S. R. Das, "Performance comparison of 3G and metro-scale WiFi for vehicular network access," in *Proc. 10th Annual Conference on Internet Measurement, IMC '10*, Melbourne, Australia, Nov. 2010.

[74] R. Gass and C. Diot, "An experimental performance comparison of 3G and Wi-Fi," in *Proc. 11th International Conference on Passive and Active Measurement, PAM'10*, Zurich, Switzerland, Apr. 2010.

[75] ——, "Eliminating backhaul bottlenecks for opportunistically encountered Wi-Fi hotspots," in *Proc. 71st IEEE Vehicular Technology Conference, VTC '10-Spring*, Taipei, Taiwan, May 2010.

[76] A. Balasubramanian, R. Mahajan, and A. Venkataramani, "Augmenting mobile 3g using WiFi," in *Proc. 8th International Conference on Mobile Systems, Applications, and Services, MobiSys '10*, San Francisco, CA, USA, Jun. 2010.

[77] T. Han and N. Ansari, "Opportunistic content pushing via WiFi hotspots," in *Proc. 3rd IEEE International Conference on Network Infrastructure and Digital Content, (IC-NIDC 2012)*, Beijing, China, Sep. 2012.

[78] H.-Y. Hsieh and R. Sivakumar, "A transport layer approach for achieving aggregate bandwidths on multi-homed mobile hosts," in *Proc. 8th Annual International Conference on Mobile Computing and Networking, MobiCom '02*, Atlanta, Georgia, USA, Sep. 2002.

[79] P. Rodriguez, R. Chakravorty, J. Chesterfield, I. Pratt, and S. Banerjee, "MAR: a commuter router infrastructure for the mobile internet," in *Proc. 2nd International Conference on Mobile Systems, Applications, and Services, MobiSys '04*, Boston, MA, USA, Jun. 2004.

[80] C.-L. Tsao and R. Sivakumar, "On effectively exploiting multiple wireless interfaces in mobile hosts," in *Proc. 2009 ACM Conference on Emerging Network Experiment and Technology, CoNEXT '09*, Rome, Italy, Dec. 2009.

[81] T.-Y. Chang, Z. Zhuang, A. Velayutham, and R. Sivakumar, "WebAccel: Accelerating web access for low-bandwidth hosts," *Comput. Netw.*, vol. 52, pp. 2129–2147, Aug. 2008.

[82] J. Grozdanovic and D. Lukovic, "Implementation of Nett-Gain in GPRS system in MTS 064," Telekom Srbija, MTS 064, Tech. Rep., 2007.

[83] "Nettgain technology-white paper," *Flash Netwoks*, 2005, white paper.

[84] R. Kokku, P. Yalagandula, A. Venkataramani, and M. Dahlin, "NPS: A non-interfering deployable web prefetching system," in *In Proc. Fourth USENIX Symposium on Internet Technologies and Systems*, Seattle, WA, Mar. 2003.

[85] A. Venkataramani, R. Kokku, and M. Dahlin, "TCP Nice: a mechanism for background transfers," *SIGOPS Oper. Syst. Rev.*, vol. 36, pp. 329–343, December 2002.

[86] J. Kim, R. A. Baratto, and J. Nieh, "pTHINC: a thin-client architecture for mobile wireless web," in *Proc. 15th International Conference on World Wide Web, WWW '06*, Edinburgh, Scotland, May 2006.

[87] I. Mohomed, J. C. Cai, and E. de Lara, "URICA: Usage-awaRe Interactive Content Adaptation for mobile devices," *SIGOPS Oper. Syst. Rev.*, vol. 40, pp. 345–358, April 2006.

[88] J. Mickens, "Silo: exploiting JavaScript and DOM storage for faster page loads," in *Proc. 2010 USENIX Conference on Web Application Development, (WebApps'10)*, Boston, MA, Jun. 2010.

[89] A. Begen, T. Akgul, and M. Baugher, "Watching video over the web, part i: Streaming protocols," *IEEE Internet Computing*, vol. 99, no. PrePrints, 2010.

[90] A. Zambelli, "IIS smooth streaming technical overview," Microsoft Corporation, Mar. 2009, white paper.

[91] "HTTP live streaming overview - networking, Internet & web," Apple Inc., 2010, white paper.

[92] "HTTP dynamic streaming on the Adobe flash platform," Adobe System Inc., 2010, white paper.

[93] R. Pantos and W. May, "HTTP live streaming (draft-pantos-http-live-streaming-05)," 2010. [Online]. Available: http://tools.ietf.org/html/draft-pantos-http-live-streaming-05

[94] "Transparent end-to-end packet-switched streaming service (PSS)," ETSI TS 126 234 V9.5.0 (2011-01);Protocols and codecs, 2011.

[95] B. Wang, J. Kurose, P. Shenoy, and D. Towsley, "Multimedia streaming via TCP: An analytic performance study," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 4, pp. 16:1–16:22, May 2008.

[96] X. Zhu and B. Girod, "Video streaming over wireless networks," in *In Proceedings of European Signal Processing Conference, (EUSIPCO)*, Poland, Sep. 2007.

[97] S. Akhshabi, A. C. Begen, and C. Dovrolis, "An experimental evaluation of rate-adaptation algorithms in adaptive streaming over HTTP," in *Proc. 2nd Annual ACM Conference on Multimedia Systems, MMSys '11*, San Jose, CA, USA, Feb. 2011.

[98] A. Goel, C. Krasic, and J. Walpole, "Low-latency adaptive streaming over TCP," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 4, pp. 20:1–20:20, September 2008.

[99] R. Kuschnig, I. Kofler, and H. Hellwagner, "An evaluation of TCP-based rate-control algorithms for adaptive internet streaming of H.264/SVC," in *Proc. 1st Annual ACM Conference on Multimedia Systems, MMSys '10*, Phoenix, Arizona, USA, Feb. 2010.

[100] M. Prangl, I. Kofler, and H. Hellwagner, "Towards QoS improvements of TCP-based media delivery," in *Proc. 4th International Conference on Networking and Services, ICNS '08.*, Gosier, Guadeloupe, Mar. 2008.

[101] K. Xu, Y. Tian, and N. Ansari, "TCP-Jersey for wireless ip communications," *IEEE J. Sel. Areas Commun.*, vol. 22, no. 4, pp. 747 – 756, May 2004.

[102] A. Argyriou, "Real-time and rate-distortion optimized video streaming with TCP," *Image Commun.*, vol. 22, pp. 374–388, April 2007.

[103] C. Krasic, J. Walpole, and W.-c. Feng, "Quality-adaptive media streaming by priority drop," in *Proc. 13th International Workshop on Network and Operating Systems Support for Digital Audio and Video, NOSSDAV '03*, Monterey, CA, USA, Jun. 2003.

[104] H. Won, H. Cai, D. Y. Eun, K. Guo, A. Netravali, I. Rhee, and K. Sabnani, "Multicast scheduling in cellular data networks," *IEEE Trans. Wireless Commun.*, vol. 8, no. 9, pp. 4540–4549, Sep. 2009.

[105] R. Kuschnig, I. Kofler, and H. Hellwagner, "Evaluation of HTTP-based request-response streams for internet video streaming," in *Proc. 2nd Annual ACM Conference on Multimedia Systems, MMSys '11*, San Jose, CA, USA, Feb. 2011.

[106] ——, "Improving internet video streamilng performance by parallel TCP-based request-response streams," in *Proc. 7th IEEE Conference on Consumer Communications and Cetworking Conference, CCNC'10*, Las Vegas, Nevada, USA, Jan. 2010.

[107] T. Nguyen and S.-C. S. Cheung, "Multimedia streaming using multiple TCP connections," in *Proc. 24th IEEE International Performance, Computing, and Communications Conference, IPCCC '05*, Phoenix, AZ, USA, Apr. 2005.

[108] R. Chakravorty, S. Banerjee, and S. Ganguly, "MobiStream: Error-resilient video streaming in wireless WANs using virtual channels," in *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, Apr. 2006, pp. 1 –14.

[109] Q. Li and M. van der Schaar, "Providing adaptive QoS to layered video over wireless local area networks through real-time retry limit adaptation," *IEEE Trans. Multimedia*, vol. 6, no. 2, pp. 278 – 290, Apr. 2004.

[110] A. Miu, J. G. Apostolopoulos, W.-t. Tan, and M. Trott, "Low-latency wireless video over 802.11 networks using path diversity," in *Proc. 2003 IEEE International Conference on Multimedia and Expo, ICME '03*, Baltimore, MD, USA, Jul. 2003.

[111] J. Chesterfield, R. Chakravorty, J. Crowcroft, P. Rodriguez, and S. Banerjee, "Experiences with multimedia streaming over 2.5G and 3G networks," in *First International Workshop on Broadband Wireless Multimedia: Algorithms, Architectures and Applications, BroadWiM '04*, San Jose, CA, Oct. 2004.

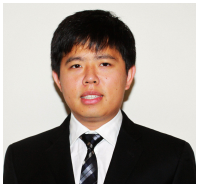
[112] J. Chesterfield, R. Chakravorty, I. Pratt, S. Banerjee, and P. Rodriguez, "Exploiting diversity to enhance multimedia streaming over cellular links," in *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, vol. 3, Mar. 2005, pp. 2020 – 2031 vol. 3.

[113] D. Kaspar, K. Evensen, P. Engelstad, A. F. Hansen, P. Halvorsen, and C. Griwodz, "Enhancing video-on-demand playout over multiple heterogeneous access networks," in *Proc. 7th IEEE Conference on Consumer Communications and Cetworking Conference, CCNC'10*, Las Vegas, Nevada, USA, Jan. 2010.

[114] D. Li, C.-N. Chuah, G. Cheung, and S. J. B. Yoo, "MUVIS: multi-source video streaming service over wlans," *J. Communication and Networks (JCN)*, vol. 7, pp. 144–156, 2005.

[115] M.-H. Lu, P. Steenkiste, and T. Chen, "Robust wireless video streaming using hybrid spatial/temporal retransmission," *IEEE J. Sel. Areas Commun.*, vol. 28, pp. 476–487, April 2010.

[116] L. De Cicco and S. Mascolo, "An experimental investigation of the akamai adaptive video streaming," in *HCI in Work and Learning, Life and Leisure*. Springer Berlin / Heidelberg, 2010, vol. 6389, pp. 447–464.

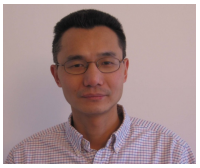


Tao Han received B.E. in Electrical Engineering and M.E. in Computer Engineering from Dalian University of Technology and Beijing University of Posts and Telecommunications, respectively. He is currently a Ph.D candidate in Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, New Jersey. His research interests include mobile and cellular networks, content delivery acceleration, network optimization, and energy efficient networking.

Nirwan Ansari [S'78-M'83-SM'94-F'09] received BSEE (summa cum laude with a perfect GPA) from NJIT, MSEE from the University of Michigan, Ann Arbor, and Ph.D. from Purdue University, West Lafayette, IN. He joined NJIT in 1988, where he is Professor of Electrical and Computer Engineering. He has also assumed various administrative positions at NJIT. He was Visiting (Chair/Honorary) Professor at several universities. His current research focuses on various aspects of broadband networks and multimedia communications.

Prof. Ansari has served on the Editorial/Advisory Board of eight journals. He was elected to serve in the IEEE Communications Society (ComSoc) Board of Governors as a member-at-large (2013-2014) as well as IEEE Region 1 Board of Governors as the IEEE North Jersey Section Chair. He has chaired ComSoc technical committees, and has been actively organizing numerous IEEE International Conferences/Symposia/Workshops, assuming leadership roles as Chair or TPC Chair at various Conferences, Symposia and Workshops.

Prof. Ansari has authored Computational Intelligence for Optimization (Springer 1997) with E.S.H. Hou, Media Access Control and Resource Allocation for Next Generation Passive Optical Networks (Springer, 2012) with J. Zhang, and edited Neural Networks in Telecommunications (Springer 1994) with B. Yuhas. He has also contributed over 400 publications, over one third of which were published in widely cited refereed journals/magazines. He has been granted over fifteen U.S. patents. He has also guest-edited a number of special issues, covering various emerging topics in communications and networking. He has been frequently selected to deliver keynote addresses, distinguished lectures, and tutorials. Some of his recent recognitions include a couple of best paper awards, several Excellence in Teaching Awards, Thomas Alva Edison Patent Award (2010), NJ Inventors Hall of Fame Inventor of the Year Award (2012), and designation as a ComSoc Distinguished Lecturer (2006-2009).



Mingquan Wu received his Ph.D. in electrical engineering from Michigan State University in 2005. From November 2005 to August 2010, he was a member of technical staff in Thomson Corporate Research. He joined Huawei as a senior researcher in September 2010. His research interests include multimedia reliable transmission over wireless networks, network modeling and resource optimization, ad hoc and overlay network transport protocol design, content delivery acceleration, etc. He has published over 20 referred papers and has more than a dozen awarded and pending patents. He has multiple proposals accepted by IEEE 802.11s, IEEE802.11aa and IEEE802.16j standards.

Heather Yu got her Ph.D. from Princeton University in 1998. Currently, she is the Director of Media Lab New Jersey Office of Futurewei Technologies, a.k.a. Huawei US R&D Center, located at Bridgewater, NJ, focusing on multimedia processing and multimedia delivery technology research. Before assuming this new role in May 2012, she was the Director of the Media Networking Lab of Futurewei which develops multimedia technologies for multiple business units within the company. With the mission of establishing a world class R&D team and leading the key multimedia technology innovations, she led the NJ team successfully accomplished the development of several new media technology research areas and a series of new technology innovations offering competitive edge capabilities and supporting various functionalities for Huawei's products. Before joining Huawei, she was with Panasonic Princeton Lab working on media communication, media processing, media security, and P2P technology research. Since graduated from Princeton, Heather served numerous positions in related associations, such as Chair of the IEEE Multimedia Communications Tech Committee, IEEE Communications Society Strategic Planning Committee member, IEEE Human Centric Communications emerging technology committee chair, Associate Editor-in-Chief for PPNA journal, AEs of several IEEE journals/magazines, and Conference chair and TPC chair for many conferences in the field. She holds 27 granted US patents and has many in pending. She published 70+ publications, including 4 books, P2P Networking and Applications, Semantic Computing, P2P Handbooks, and Multimedia Security Technologies for Digital Rights Management.