

VIDEO RATE ADAPTATION TO REVERSE LINK CONDITIONS

TECHNICAL FIELD

[0001] The disclosure relates to video encoding and, more particularly, techniques for adapting a video encoding rate to reverse link conditions.

BACKGROUND

[0002] A cellular phone may include an audio capture device, such as a microphone or speech synthesizer, and an audio encoder to generate audio packets (or frames). The phone may use communication protocol layers, such as radio link protocol (RLP), medium access control (MAC), and physical (PHY) layers. The phone may place audio packets in a RLP queue. A MAC layer module may generate MAC layer packets from contents of the RLP queue. The MAC layer packets may be converted to PHY layer packets for transmission across a communication channel to another communication device.

SUMMARY

[0003] One aspect relates to a method comprising: estimating video throughput based on a size of a video flow radio link protocol (RLP) queue at an access terminal; and encoding video data using the estimated video throughput.

[0004] Another aspect relates to a method comprising: determining a first size V_n of a video queue in a radio link protocol (RLP) layer at a first time t_n based on a video frame rate; determining a second size V_m of the video queue at a second time t_m based on an audio frame rate; if the first size V_n or the second size V_m is greater than zero, then using the first size V_n , a previous size V_{n-1} of the video queue associated with a previous video frame, a previous video frame size B_{n-1} , the first time t_n , and a time t_{n-1} associated with the previous size of the video queue to determine a video throughput VTP; if the first size V_n and the second size V_m are equal to zero, then searching for an earlier time based on the audio frame rate when the video queue size was greater than zero; after finding the earlier time based on the audio frame rate when the video queue size was greater than zero, using an earlier queue size V_{m-i} based on the audio frame rate, the

previous size V_{n-1} of the video queue associated with the previous video frame, the previous video frame size B_{n-1} , the earlier time t_{m-1} , and the time t_{n-1} associated with the previous size of the video queue to determine a video throughput VTP; using the determined video throughput VTP to determine a channel-constrained video frame size; and using the channel-constrained video frame size to control a video encoding rate.

[0005] Another aspect relates to a method comprising: determining a size of a video queue in a radio link protocol (RLP) layer; determining a power headroom limitation from a medium access control (MAC) layer; using the determined power headroom limitation to determine a MAC payload size; using the determined MAC payload size and an estimate of how many transmission opportunities are given to video in a time period to determine video throughput; using the determined video throughput and the determined size of the video queue in the RLP layer to determine a channel-constrained video frame size; and using the channel-constrained video frame size to control a video encoding rate.

[0006] Another aspect relates to an apparatus comprising a machine-readable memory that stores a set of instructions. The instructions are configured to: determine a first size V_n of a video queue in a radio link protocol (RLP) layer at a first time t_n based on a video frame rate; determine a second size V_m of the video queue at a second time t_m based on an audio frame rate; if the first size V_n or the second size V_m is greater than zero, then use the first size V_n , a previous size V_{n-1} of the video queue associated with a previous video frame, a previous video frame size B_{n-1} , the first time t_n , and a time t_{n-1} associated with the previous size of the video queue to determine a video throughput VTP; if the first size V_n and the second size V_m are equal to zero, then searching for an earlier time based on the audio frame rate when the video queue size was greater than zero; after finding the earlier time based on the audio frame rate when the video queue size was greater than zero, use an earlier queue size V_{m-i} based on the audio frame rate, the previous size V_{n-1} of the video queue associated with the previous video frame, the previous video frame size B_{n-1} , the earlier time t_{m-1} , and the time t_{n-1} associated with the previous size of the video queue to determine a video throughput VTP; use the determined video throughput VTP to determine a channel-constrained video frame size; and use the channel-constrained video frame size to control a video encoding rate.

[0007] Another aspect relates to an apparatus comprising a machine-readable memory that stores a set of instructions. The instructions are configured to: determine a size of

a video queue in a radio link protocol (RLP) layer; determine a power headroom limitation from a medium access control (MAC) layer; use the determined power headroom limitation to determine a MAC payload size; use the determined MAC payload size and an estimate of how many transmission opportunities are given to video in a time period to determine video throughput; use the determined video throughput and the determined size of the video queue in the RLP layer to determine a channel-constrained video frame size; and use the channel-constrained video frame size to control a video encoding rate.

[0008] Another aspect relates to an apparatus comprising: a radio link protocol (RLP) layer queue configured to store video data; a first unit configured to receive a size of the RLP video queue and a power headroom limitation from a medium access control (MAC) layer, use the power headroom limitation to determine a MAC payload size, use the determined MAC payload size and an estimate of how many transmission opportunities are given to video in a time period to determine video throughput, and use the determined video throughput and the size of the video queue in the RLP layer to determine a channel-constrained video frame size; a second unit to use the channel-constrained video frame size to control a video encoding rate; and a video encoder to use the video encoding rate to encode video.

[0009] Another aspect relates to apparatus comprising: means to determine a size of a video queue in a radio link protocol (RLP) layer; means to determine a power headroom limitation from a medium access control (MAC) layer; means to use the determined power headroom limitation to determine a MAC payload size; means to use the determined MAC payload size and an estimate of how many transmission opportunities are given to video in a time period to determine video throughput; means to use the determined video throughput and the determined size of the video queue in the RLP layer to determine a channel-constrained video frame size; and means to use the channel-constrained video frame size to control a video encoding rate.

[0010] The details of one or more embodiments are set forth in the accompanying drawings and the description below.

BRIEF DESCRIPTION OF DRAWINGS

[0011] Fig. 1 illustrates a video encoding and decoding system.

[0012] Figs. 2A and 2B illustrate simulated data showing increased video delay when reverse link (RL) channel conditions are poor.

[0013] Fig. 3 illustrates a correlation between (a) video delay of each frame and (b) lower-layer information.

[0014] Fig. 4 illustrates a first rate adaptation technique with examples of structures and data flows.

[0015] Fig. 5A illustrates a frequency of application layer inquiries to a lower layer to get video flow RLP queue size, where the frequency is based on an audio frame rate and a video frame rate.

[0016] Figs. 5B-5E illustrate examples of determining RLP queue size.

[0017] Fig. 6 illustrates a second rate adaptation technique with examples of structures and data flows.

[0018] Figs. 7A-7B illustrate tables to convert power headroom limitation to maximum payload size.

DETAILED DESCRIPTION

[0019] Fig. 1 illustrates a video encoding and decoding system 10. The system 10 includes an encoder system 12 sending data across a transmission channel 16 to a decoder system 14. The encoder system 12 may be in a first video communication device and may include an audio source 17, video source 18, video encoder 20, audio encoder 22, real-time transport protocol (RTP)/user datagram protocol (UDP)/ Internet protocol (IP)/ point-to-point protocol (PPP) conversion module 26, radio link protocol (RLP) queue 28, MAC layer module 30 and physical (PHY) layer module 32. Other embodiments of the encoder system 12 may include other elements instead of or in addition to the elements shown in Fig. 1. Other embodiments of the encoder system 12 may include fewer elements than those shown in Fig. 1.

[0020] The decoder system 14 may be in another video communication device and may include a PHY layer module 34, MAC layer module 36, RLP queue 38, RTP/UDP/IP/PPP conversion module 40, video decoder 42, audio decoder 44, audio output unit 46 and video output unit 48. Other embodiments of the decoder system 14 may include other elements instead of or in addition to the elements shown in Fig. 1.

Other embodiments of the decoder system 14 may include fewer elements than those shown in Fig. 1.

[0021] System 10 may provide bi-directional video and audio transmission, e.g., for video telephony (VT) via transmission channel 16. Reciprocal encoding, decoding, and conversion modules may be provided on opposite ends of channel 16. In some embodiments, encoder system 12 and decoder system 14 may be embodied within video communication devices such as wireless mobile terminals equipped for video streaming, VT, or both. The mobile terminals may support VT according to packet-switched standards such as RTP, UDP, IP, or PPP.

[0022] RTP/UDP/IP/PPP conversion module 26 adds appropriate RTP/UDP/IP/PPP header data to audio and video data received from audio encoder 22 and video encoder 20, and places the data in RLP queue 28. RTP runs on top of UDP, while UDP runs on top of IP, and IP runs on top of PPP. MAC layer module 30 generates MAC RLP packets from the contents of RLP queue 28. PHY layer module 32 converts the MAC RLP packets into PHY layer packets for transmission over channel 16.

[0023] PHY layer module 34 and MAC layer module 36 of decoding system 14 operate in a reciprocal manner. PHY layer module 34 converts PHY layer packets received from channel 16 to MAC RLP packets. MAC layer module 36 places the MAC RLP packets into RLP queue 38. RTP/UDP/IP/PPP conversion module 40 strips the header information from the data in RLP queue 38, and reassembles the video and audio data for delivery to video decoder 42 and audio decoder 44, respectively.

[0024] System 10 may be designed to support one or more wireless communication technologies such as code division multiple access (CDMA), frequency division multiple access (FDMA), time division multiple access (TDMA), or orthogonal frequency divisional multiplexing (OFDM), or another suitable wireless technique. The above wireless communication technologies may be delivered according to any of a variety of radio access technologies. For example, CDMA may be delivered according to cdma2000 or wideband CDMA (WCDMA) standards. TDMA may be delivered according to the Global System for Mobile Communications (GSM) standard. The Universal Mobile Telecommunication System (UMTS) standard permits GSM or WCDMA operation. For VT applications, system 10 may be designed to support high data rate (HDR) technologies such as cdma2000 1x EV-DO, Release 0, Revision A or subsequent EV-DO releases.

[0025] The video source 18 may be a video capture device, such as one or more video cameras, one or more video archives, or a combination of video cameras and video archives. The video encoder 20 generates encoded video data according to a video compression method, such as MPEG-4. Other video compression methods may be used, such as the International Telecommunication Union (ITU) H.263, ITU H.264, or MPEG-2 methods. Video encoder 20 may provide a video source rate control scheme that is generally CODEC-dependent. For example, video encoder 20 may be adapted for video encoding according to MPEG4, ITU H.263 or ITU H.264. Video encoder 20 may be implemented by a DSP or embedded logic core.

[0026] The audio source 17 may be an audio capture device, such as a microphone, or a speech synthesizer device. The audio encoder 22 may encode audio data to accompany the video data. The audio data may be encoded according to an audio compression method, such as adaptive multi-rate narrow band (AMR-NB), or other techniques. For VT applications, the video will permit viewing of a party to a VT conference, and the audio will permit the speaking voice of that party to be heard.

[0027] In operation, RTP/UDP/IP/PPP conversion module 26 obtains video and audio data packets from video encoder 20 and audio encoder 22. RTP/UDP/IP/PPP conversion module 26 adds appropriate header information to the audio packets and inserts the resulting data within RLP queue 28. Likewise, RTP/UDP/IP/PPP conversion module 26 adds appropriate header information to the video packets and inserts the resulting data within RLP queue 28. MAC layer module 30 retrieves data from RLP queue 28 and forms MAC layer packets. Each MAC layer packet carries RTP/UDP/IP/PPP header information and audio or video packet data that is contained within RLP queue 28.

[0028] Audio packets may be inserted into RLP queue 28 independently of video packets. In some cases, a MAC layer packet generated from the contents of RLP queue 28 will carry only header information and video packet data. In other cases, the MAC layer packet will carry only header information and audio packet data.

[0029] In some cases, the MAC layer packet will carry header information, audio packet data and video packet data, depending on the contents of RLP queue 28. The MAC layer packets may be configured according to a radio link protocol (RLP), and may be referred to as MAC RLP packets. PHY layer module 32 converts the MAC RLP audio-video packets into PHY layer packets for transmission across channel 16.

[0030] Channel 16 carries the PHY layer packets to decoder system 14. For example, channel 16 may be a wired connection, such as a local or wide-area wired network. Alternatively, as described herein, channel 16 may be a wireless channel such as a cellular, satellite or optical channel.

[0031] Channel conditions may be a concern for wired and wireless channels, but are especially problematic for mobile VT applications performed over a wireless channel 16, in which channel conditions may suffer due to fading or congestion. For example, channel 16 may be characterized by a reverse link (RL) having a throughput that varies according to channel conditions. Throughput may be estimated based on channel conditions, as represented by one or more of current wireless channel transmit rate, wireless base station activity, and transmit power limitations. For example, the channel conditions may be determined based on current MAC layer data rate, a reverse activity bit (RAB), and a power amplifier (PA) limit.

[0032] Video encoder 20 may maintain a virtual video buffer representing an amount of the encoded video relative to a target encoding rate. The target encoding rate may be a maximum encoding rate specified for video packets transmitted over channel 16. Video encoder 20 may control an actual encoding rate of the video from video source 18.

[0033] PHY layer module 34 of decoder system 14 identifies the MAC layer packets from the PHY layer packets and reassembles the content into MAC RLP packets. MAC layer module 36 then reassembles the contents of the MAC RLP packets to provide video and audio packets for insertion within RLP queue 38. RTP/UDP/IP/PPP module 40 removes the accompanying header information and provides video packets to video decoder 42 and audio packets to audio decoder 44.

[0034] Video decoder 42 decodes the video data frames to produce a stream of video data for use in driving a display device. Audio decoder 44 decodes the audio data to produce audio information for presentation to a user, e.g., via an audio speaker.

[0035] Video telephony (VT) refers to real-time communication of packets carrying audio and video data between at least two devices, such as systems 12 and 14. A first VT device 12 includes a video encoder 20 that obtains video from a video capture device 18, such as a video camera or video archive, and generates video packets. Similarly, an audio encoder 22 in the VT device 12 obtains audio from an audio capture device 17, such as a microphone or speech synthesizer, and generates audio packets. The video packets and audio packets are placed in a RLP queue 28. A MAC layer

module 30 generates MAC layer packets from the contents of the RLP queue 28. The MAC layer packets are converted to PHY layer packets for transmission across a communication channel 16 to a second VT device 14.

[0036] In mobile VT applications, a VT device (wireless terminal) receives PHY layer packets via a wireless forward link (FL) (i.e., “downlink”) from a base station. A VT device transmits PHY layer packets via a wireless reverse link (RL) (i.e., “uplink”) to a base station. Each VT device includes PHY and MAC layers to convert the received PHY and MAC layer packets and reassemble the packet payloads into audio packets and video packets. A video decoder 42 within the VT device decodes the video data for presentation to a user via a display device (video output) 48. An audio decoder 44 within the VT device decodes the audio data for output via an audio speaker (audio output) 46.

[0037] Mobile VT in a wireless environment may be challenging because the data rate over the wireless channel may be limited and may vary with time. For example, in a CDMA2000 1x EV-DO Release 0 or Revision A network, the data rate may vary due to channel conditions within a wireless coverage area or traffic congestion among multiple VT users. Channel conditions, excessive video content, or both can cause significant delays in transmission of video. For example, when RL throughput is reduced, video transmission can overwhelm the RL and increase video transmission delay. As a result, mobile VT can be susceptible to undesirable video and/or audio delay, which undermines the ability to provide smooth video conferencing in real-time.

[0038] The description below provides techniques for video rate adaptation (controlling the encoding rate of source video) for applications, such as VT, to reduce video delay over a range of channel conditions. The video source rate adaptation may be called channel-adaptive. The techniques may be effective in reducing degradation of spatial and temporal quality when the video source encoding rate is reduced due to channel conditions or excessive video content or complexity.

[0039] Performance of video source encoding rate control can be evaluated by end-to-end delay, which is delay of video transmission between a sender and a recipient, e.g., in a mobile wireless VT system. End-to-end delay may include buffering and transmission delays, spatial visual quality, number of skipped video frames, encoder buffer underflow, which indicates bandwidth underutilization, encoder buffer overflow, which causes frame skipping, decoder buffer underflow, which indicates there is no data

to decode and less display refresh, decoder buffer overflow, which indicates lost data, receiver display refresh rate, audio-video synchronization, encoder-side peak signal to noise ratio (PSNR), and initial buffer delay following a first intra (I) frame.

[0040] Video telephony (VT) may be an important application for CDMA2000 1x EV-DO Rev A networks. EV-DO Rev A may provide data rates up to 3.1Mbps on forward link (FL) (downlink) and 1.8Mbps on RL (uplink). EV-DO Rev A also supports intra- and inter-user quality of service (QoS). Intra-user QoS gives audio data higher priority than video data, which reduces audio delay by trading off video delay when channel conditions degrade. Compared to an EV-DO Release 0 network, EV-DO Rev A's more symmetric, higher data rates and QoS support may be more suitable to carry bandwidth-demanding, delay-sensitive traffic and may enhance overall VT quality.

[0041] Although an EV-DO Rev A network provides unique features that accommodate VT traffic, one challenging problem may be excessive video delay when underlying channel conditions become poor. This usually happens when a VT mobile device user experiences faded channels or moves to an edge of a sector and becomes headroom limited. Because intra-user QoS is supported, audio will be served and transmitted with a higher priority than video. There may even be some moments when there is no bandwidth to transmit any video. As a result, video data will be queued in a buffer until the resource is freed up from audio data or after channel conditions improve.

[0042] Figs. 2A and 2B illustrate simulated data showing increased video delay when RL channel conditions are poor. The simulation sends 48 kbps, 15 frames per second (fps) MPEG-4 compressed video and enhanced variable rate coder (EVRC) encoded audio with 3-frame bundling over EV-DO Rev A RL channel emulators. The FL can also cause additional video delay, but the FL problem is independent from the RL. The RL rate adaptation techniques described below may help improve overall end-to-end video delay.

[0043] Fig. 2A shows different channel conditions in the simulation. In this simulation, the network load is light, i.e., there are few users in the same sector. In this simulation, the MAC layer design for VT flows does not react to sector loading. This means that RL resource allocation will guarantee 48 kbps video transmission, unless the access terminal is power headroom limited. When power headroom is limited (or RL condition gets poor), the MAC layer 30 will transmit audio data before video data. Different channel conditions are simulated by different slow fading situations in Fig. 2A.

Condition 2 in Fig. 2A uses Channel-A time-varying shadow, which simulates an access terminal (AT) slowly moving about 3 kilometers per hour (kmph). When the AT moves into fading areas, it will tend to stay there for a longer period of time, as compared to conditions 4 and 6, where the AT moves faster at about 10 kmph and 120 kmph, respectively. The simulations were also done with and without soft handoff. Typically, the channel conditions without soft handoff are worse than those with soft handoff.

[0044] Fig. 2B shows audio and video delays when audio and video are transmitted over all the test channel conditions of Fig. 2A. As Fig. 2 B shows, the audio delay does not increase much for all the channel conditions. This is because the QoS supported by an EV-DO Rev A reverse link provides priority transmission for audio data over video. When RL bandwidth decreases, RL will allocate available resources to audio data first and then allocate the remaining resources to video data transmission. When the RL condition is poor, audio data uses most of the resources (or bandwidth), while video data will be buffered in a transmission queue (or video flow RLP queue 28). As a result, video delay increases dramatically as shown in Fig. 2B.

[0045] The simulation collected data by sending audio and video for five minutes. The delay is measured for each video frame. For conditions 2 through 5, the video has long delays up to 2 seconds on average. The cumulative delay distribution at 95% can be as high as up to 12.5 seconds. These values are the results of five minutes of experiment time, assuming that the RLP queue has unlimited physical memory to store video data.

[0046] If the time was increased to 20 minutes, these values would even be worse. That is, the video flow RLP queue will grow large because the AT is unable to sustain 48 kbps video transmission. This is unacceptable for real-time VT applications. The main reason is that the video encoder 20 is not aware of channel degradation and continues to produce 48 kbps video to send across the RL. Since the RL cannot support video data at such a high rate during channel fading, most of the video data will be buffered. This buffering causes delay.

[0047] Therefore, it is highly desirable to adjust the video encoding rate to match what the RL can support to avoid any video data being buffered to reduce video delay. A video delay target at 95% may be 200 ms.

[0048] The description below describes new video rate adaptation techniques for video telephony applications, such as video conferencing and video sharing. These techniques are described with a CDMA2000 EV-DO Rev A network, but other types of networks

may be used. These techniques address the problem of increased video delay when channel conditions deteriorate on the RL.

[0049] One proposed method is a cross-layer optimization design that takes characteristics of EV-DO Rev A RL into account and uses information from the MAC layer 30 and RLP layer 28 passed to the video encoder 20. The video encoder 20 uses this information to monitor RL conditions and adjust its encoding rate according to the channel condition.

[0050] Simulation results show that the method can effectively reduce average video delay by up to 94%, and the 95 percentile delay (delay value within which the decoder 42 will receive 95% of the video packets transmitted by the encoder 20) can be improved by up to 98% for different channel conditions, assuming that the RLP queue has unlimited physical memory to store video data. In addition, the effective video throughput can be increased by up to 4 kilobits per second (kbps). The computational complexity of the proposed method may be low, so it can be implemented in computation-limited mobile devices.

[0051] **Correlation between video delay and lower-layer information**

[0052] Fig. 3 illustrates a correlation between (a) video delay (in milliseconds and divided by 100) of each frame according to the time when the frame was generated and (b) lower-layer information, e.g., video flow RLP queue size (in bytes and divided by 100) from RLP layer and power headroom limitation from MAC layer. Power headroom limitation is measured in decibels (dBs) and limits the maximum possible payload size of MAC layer packets. The lower the power headroom limitation value is, the smaller the maximum possible payload size is, and hence the lower the throughput. The headroom limitation may indicate the maximum rate that is allowed to be used in transmission, based on the current transmit power. The PA limit represents transmit power headroom and indicates when channel conditions have degraded.

[0053] Fig. 3 shows a strong correlation between video flow RLP queue size and video delay. When the RLP queue size increases, video delay also increases, such as times around 2.62, 2.8 and 2.9 along the x-axis in Fig. 3.

[0054] There is also a strong correlation between power headroom limitation and video delay. When the power headroom limitation is below 10 dB, as circled in Fig. 3, the video delay is increased. The lower the power headroom limitation is, the larger the

video delay is. Based on these observations, video flow RLP queue size and power headroom limitation seem to be very useful information for video rate adaptation.

[0055] Video Rate Adaptation Using EV-DO Rev A RL MAC Parameters

[0056] Two different video rate adaptation techniques are described to reduce video delay. Both methods use information from lower layers such as the MAC and RLP. Either technique can be applied with different delay performance, depending on what information can be obtained from the lower layers.

[0057] The first technique is based solely on video flow RLP queue size. The second technique is an enhanced version based on both video flow RLP queue size and power headroom limitation. The second technique addresses drawbacks of the first technique and has less (it is less because more reliable information is available in the second enhanced approach so that it does not need to do as much as in the first approach to try to get accurate video throughput estimation) computation complexity but better delay performance without sacrificing effective video throughput.

[0058] One may implement either technique based which information is available without waiting for all information to be ready. If more information is available, the second technique may improve delay performance further. In general, rate adaptation may use all possible information from lower layers, in addition to video flow queue size and power headroom limitation. More MAC information may be passed to the video encoder 20 in order to allow more accurate and flexible rate adaptation. The description below focuses on how to use queue size and power headroom limitation as examples to do rate adaptation.

[0059] First Rate Adaptation Technique

[0060] There may be two different constraints for rate adaptation. A first possible constraint may be bit-rate constraint, which may guarantee 48-Kbps video data rate even when the channel 16 can afford higher rates. Some systems or networks may not have this bit-rate constraint. A second constraint is channel constraint, which will limit the video rate based on the current channel conditions.

[0061] Fig. 4 illustrates a first rate adaptation technique with examples of structures and data flows. A video encoder 400 encodes video data from a video source 401 using a video encoding rate from video rate control unit 402. The video encoder 400 sends encoded video data to a RTP/UDP/IP/PPP unit 406, which sends video packets to the video flow RLP queue 410. The video flow RLP queue 410 sends video packets to the

reverse traffic channel MAC (RTCMAC) 412. The RTCMAC implements a protocol to provide procedures followed by a communication device to transmit over the RL of channel 16.

[0062] A large video flow RLP queue size may indicate that video data has a long delay. The video encoder 400 and/or other units in Fig. 4 may frequently monitor video flow RLP queue size and adjust the video encoding rate when necessary.

[0063] One way to adjust video rate is to look at the instantaneous video RLP queue size and skip one frame when the size exceeds a threshold. This approach may incur too many skipped frames and may not provide graceful video quality degradation. A better approach is to look at first-order statistics of video RLP queue size to estimate the available video throughput, which can be used to determine the current frame size.

[0064] The video encoder 400 may allocate bits for each frame (i.e., determine frame size) on a frame-by-frame basis. For instance, for 15-fps video, the frame size info unit 404 may decide the size of each frame every 66 ms. If the RL is not power headroom limited, and the sector is not loaded, the frame size info unit 404 allocates frame size based on a target bit rate constraint 414. Otherwise, the channel constrained frame size estimation unit 408 needs to know the amount of data to generate and not overwhelm the RL. Therefore, the channel constrained frame size estimation unit 408 estimates a maximum amount of data that the RL can handle between two consecutive video frames. An estimation error will be reflected in video flow RLP queue size, and this will be taken into account when the channel constrained frame size estimation unit 408 decides the frame size of the next frame.

[0065] Step 1: Get video flow RLP queue size

[0066] The channel-constrained frame size estimation unit 408 will query the RLP layer periodically to retrieve video flow RLP queue size V .

[0067] Fig. 5A illustrates a frequency of application layer inquiries to the RLP layer to get video flow RLP queue size, where the frequency is based on audio frame rate (every 20 ms) and video frame rate (every 66 ms if encoded at 15 fps). The video frame rate in Fig. 5A is a time period for the video encoder 400 to encode a frame $n-1$ of size B_{n-1} .

The channel-constrained frame size estimation unit 408 may separate information queried based on an audio timer and a video timer. When RLP queue size is queried based on the audio timer, the channel-constrained frame size estimation unit 408 may

record/store the queue size and the queried time as (V_m, t_m) . The audio timer is normally every 20 ms but may be 10 ms or 30 ms, depending on what speech encoder is used.

[0068] When RLP queue size is queried based on the video timer, the channel-constrained frame size estimation unit 408 may record/store the queue size and the queried time as (V_n, t_n) , as illustrated in Fig. 5A. The video timer is the time interval between two consecutive frames and is based on frame rate. For 15-fps video frame rate, the video timer is every 66 ms.

[0069] If the channel-constrained frame size estimation unit 408 only queries RLP queue size using the video timer, V_n could be zero, and the channel-constrained frame size estimation unit 408 would not know when the RL finishes transmission in the queue 410 during t_n and t_{n-1} . Thus, channel throughput may be under-estimated. This is a reason for using the audio timer to query RLP queue size more frequently. By allowing more frequent queries to the RLP layer, the channel-constrained frame size estimation unit 408 can keep track when the RLP queue 410 becomes empty.

[0070] A method for finding V_x and t_x may be expressed as:

```

if  $(V_n > 0)$  or  $(V_m > 0)$ 
     $V_x = V_n$ ;  $t_x = t_n$ ;
else
     $i=1$ ;
    loop
    {
        if  $((t_{m-i} > t_{n-1}) \text{ and } (V_{m-i} > 0)) \text{ or } (t_{m-i} \leq t_{n-1})$ 
        {
             $V_x = V_{m-i+1}$ ;  $t_x = t_{m-i+1}$ ;
            Done and stop the loop;
        }
        else  $i=i+1$ ;
    }

```

[0071] (V_x, t_x) will be used to calculate video throughput, as described below. Fig. 5A shows only 3 queries based on audio timer. This is based on the frame rate of 15 fps. If a lower frame rate such as 7.5 fps or 5 fps is used, there will be more queries, and the process above should be modified to be more generic. The method above is for searching when RLP video queue 410 becomes zero if V_n is zero. If V_n is not zero, the channel constrained frame size estimation unit 408 will use t_n as the time t_x and estimate the video throughput (described below) because the channel cannot flush out data that was generated at time t_{n-1} .

[0072] Figs. 5B-5E illustrate examples of determining RLP queue size using the method above. In Fig. 5B, the method determines if $(V_n > 0)$ or $(V_m > 0)$. Since $V_m > 0$, the method sets $V_x = V_n$ and $t_x = t_n$ because t_n is the time when the RLP queue size becomes zero. This means the channel consumes 550 bytes in between t_{n-1} and t_n .

[0073] In Fig. 5C, the method determines if $(V_n > 0)$ or $(V_m > 0)$. The answer is no, which means the RLP queue size becomes zero earlier than V_m . In this case, the method cannot set $V_x = V_n$; $t_x = t_n$. The method needs to find when the RLP queue size becomes zero. The method searches from V_m to all earlier times V_{m-1} , V_{m-2} to find when RLP queue size becomes zero. The method sets $i = 1$. The method determines whether V_{m-i} (where $i = 1$) is equal to zero. The answer is no because V_{m-i} is 150 bytes. Then the method knows t_m is the time when RLP queue size becomes zero. Then the method can set $V_x = V_m$ and $t_x = t_m$ (where $m-i+1 = m$ in this case).

[0074] In Fig. 5D, the method determines if $(V_n > 0)$ or $(V_m > 0)$. The answer is no, which means the RLP queue size becomes zero earlier than V_m . In this case, the method cannot set $V_x = V_n$; $t_x = t_n$. The method needs to determine when the RLP queue size becomes zero. The method searches from V_m to all earlier times V_{m-1} , V_{m-2} to find when RLP queue size becomes zero. The method sets $i = 1$. The method determines whether V_{m-i} (where $i = 1$) is equal to zero. The answer is yes, and the method increases i by 1. Now $i = 2$. The method determines whether V_{m-i} (where $i = 2$) is equal to zero. The answer is no, V_{m-i} is 250 bytes. Then the method knows t_{m-1} (where $m-i+1 = m-1$) is the time when RLP queue size becomes zero. Then, the method can set $V_x = V_{m-1}$; $t_x = t_{m-1}$.

[0075] In Fig. 5E, the method determines if $(V_n > 0)$ or $(V_m > 0)$. The answer is no, which means the RLP queue size becomes zero earlier than V_m . In this case, the method cannot set $V_x = V_n$; $t_x = t_n$. The method needs to determine when the RLP queue size becomes zero. The method searches from V_m to all earlier times V_{m-1} , V_{m-2} to find when RLP queue size becomes zero. The method sets $i = 1$. The method determines whether V_{m-i} (where $i = 1$) is equal to zero. The answer is yes, and the method increases i by 1. Now $i = 2$. The method determines whether V_{m-i} (where $i = 2$) is equal to zero. The answer is yes, and the method increases i by 1. Now $i = 3$. Now t_{m-i} is earlier than t_{n-1} . This means the method has checked all the RLP queue sizes in between two video frames. In this case, the method will set $V_x = V_{m-2}$; $t_x = t_{m-2}$ (where $m-i+1 = m-2$).

[0076] In this method that uses RLP queue size, the basic idea of determining V_x and t_x is to search when the RLP queue size becomes zero, if V_n is zero. This is because if V_n is zero, the method does not know when the RL transmits all the data and hence the estimation may under-estimate the channel bandwidth. If V_n is not zero, the method can simply use t_n as the time and estimate the video throughput.

[0077] Step 2: Estimate video throughput

[0078] Video throughput (VTP) since the last time a frame has been sent at time t_{n-1} to the current frame at time t_n before encoding can be estimated by the channel constrained frame size estimation unit 408 as follows:

$$VTP = \frac{B_{n-1} + V_{n-1} - V_x}{t_x - t_{n-1}},$$

[0079] where B_{n-1} is the size of frame $n-1$. t_x is the time when the video flow RLP queue size becomes empty or equals to t_n if V_n is not zero.

[0080] Step 3: Determine channel-constrained maximum frame size

[0081] After video throughput VTP is estimated, the channel-constrained frame size estimation unit 408 determines the maximum frame size that the RL can afford, assuming that the channel does not change, as follows:

$$\tilde{B}_n^{Ch} = VTP \times \frac{1}{F} - V_n + A(VTP),$$

[0082] where F is the frame rate. $A(VTP)$ is an adjusting factor to control how much video encoder 400 reacts to the channel. $A(VTP)$ may be a function of the estimated video throughput. $A(VTP)$ is useful where the previous video frame was small, and VTP underestimates the true bandwidth of the channel 16. $A(VTP)$ can also be used to control video delay according to different delay constraints from different applications. For example, video conferencing may use smaller values of $A(VTP)$, while a video sharing application may use larger values. $A(VTP)$ may be a constant or a variable. An example of $A(VTP)$ is 100 bytes.

[0083] Step 4: Determine target frame size

[0084] The frame size information unit 404 may determine the target frame size for frame n as follows:

$$\hat{B}_n = \begin{cases} \min(\tilde{B}_n^{Ch}, \tilde{B}_n^{Vb}), & \text{if } \tilde{B}_n^{Ch} > B_{\min}, \\ 0 & \text{otherwise} \end{cases}$$

[0085] where $\tilde{B}^{vb}$ is the frame size constrained by virtual buffer 416 with size W_n , which is used to control the averaged video bit rate. The virtual buffer 416 may be implemented by memory and software executed by a processor. B_{min} is the minimum frame size that is controlled to guarantee a good image quality.

[0086] If the channel constrained frame size is too small, the frame size info unit 404 will skip this frame instead of generating a bad image and wasting bandwidth.

[0087] The frame size information unit 404 sends the target frame size to the video rate control unit 402 to determine a video rate. Video rate is equal to video data frame size per time period (e.g., 66 ms).

[0088] Although the first rate adaptation technique can successfully reduce video delay, a possible drawback is that it reacts only after the channel degradation has caused video data buffered in the RLP queue 410. This technique may still create a long delay for the frame that is already buffered in the RLP queue 410.

[0089] **Second Rate Adaptation Technique**

[0090] Fig. 6 illustrates a second rate adaptation technique with examples of structures and data flows. In the second rate adaptation technique, a channel-constrained video throughput estimation unit 608 uses additional information (e.g., power amplifier (PA) headroom limitation) from the MAC layer 612 to estimate video throughput. This second technique proactively estimates (or predicts) video throughput based on power headroom limitation, instead of relying on first-order statistics of video flow RLP queue size. The MAC layer 612 uses power headroom limitation as a limiting factor to determine the payload size of MAC packets. The channel-constrained video throughput estimation unit 608 could also use the power headroom limitation to estimate video throughput because the power headroom limitation typically does not change significantly between two consecutive video frames.

[0091] Alternatively, if the unit 608 knows the instantaneous MAC payload size before encoding a frame, the unit 608 could use the instantaneous MAC payload size to predict the video throughput. In this way, the unit 608 can determine the frame size more accurately in a proactive way to avoid sending excessive data that overwhelms RL.

[0092] **Step 1: Get power headroom limitation and video flow queue size**

[0093] The channel-constrained video throughput estimation unit 608 queries the RLP layer to get video flow queue size and queries the MAC layer 612 to get power

headroom limitation. The query frequency can be every 66 ms. It is not necessary to use an audio timer in this second rate adaptation technique. The unit 608 may store power headroom limitation and video flow RLP queue size.

[0094] Step 2: Get MAC payload size based on table look-up

[0095] The MAC layer 612 may determine the MAC payload size by table look-up. The MAC layer 612 may maintain a table as shown in Fig. 7A and/or Fig. 7B to convert power headroom limitation to maximum payload size. This conversion can also be used to determine the maximum video throughput in the next step. This conversion may be done in the MAC layer 612 itself, i.e., the MAC layer 612 may pass power headroom limited payload size to the application layer. Fig. 7A illustrates a conversion table for power headroom limitation (the left-most column) and MAC payload size. The effective data rate is also shown with different termination targets.

[0096] The table in Fig. 7A is one example. There may be other tables used in MAC to decide the payload size, depending on termination targets and transmission mode (high capacity (HiCap) or low latency (LoLat)). Fig. 7B illustrates other payload size conversions based on power headroom limitation.

[0097] Step 3: Estimate video throughput

[0098] The channel-constrained video throughput estimation unit 608 can predict video throughput as follows:

$$VTP = \text{Payload_Size} * TX_Opportunities$$

[0099] *TX_Opportunities* is an estimate of how many transmission opportunities are given to video in every 66 ms. *TX_Opportunities* is determined by taking into account the current MAC design, RL characteristics such as 3-interlace structure, and audio data. For example, *TX_Opportunities* may be 3, but other values may be used.

[00100] Step 4: Determine channel-constrained maximum frame size

[00101] The channel-constrained video throughput estimation unit 608 can determine channel constrained maximum frame size with the following equation:

$$\tilde{B}_n^{Ch} = VTP - V_n + A(VTP),$$

[00102] Step 5: Determine target frame size

[00103] The frame size info unit 604 may determine the target size for the current frame.

$$\hat{B}_n = \begin{cases} \min(\tilde{B}_n^{Ch}, \tilde{B}_n^{Vb}), & \text{if } \tilde{B}_n^{Ch} > B_{\min}, \\ 0, & \text{otherwise,} \end{cases}$$

[00104] The second rate adaptation technique has lower video delay with comparable video throughput. This means using power headroom limitation can better estimate video throughput. The second rate adaptation technique may meet a delay target of 200 ms at 95% for many cases.

[00105] The techniques described in this disclosure may be implemented within a general purpose microprocessor, digital signal processor (DSP), application specific integrated circuit (ASIC), field programmable gate array (FPGA), or other equivalent logic devices. For example, video encoder system 12, and its components and modules, may be implemented as parts of an encoding process, or coding/decoding (CODEC) process, running on a digital signal processor (DSP), microprocessor, embedded core, or other processing device. Accordingly, components described as modules may form hardware components or programmable features of an encoding process, or a separate process.

[00106] In some embodiments, encoding functions may be divided among different hardware components. For example, frame-level rate control may be performed in an embedded logic core, and MB-level rate control may be performed in a DSP. As an illustration, given a target bit rate (R Kbps) and a frame rate (F fps), frame-level rate control within the embedded logic core may involve updating rate control model parameters, e.g., rho domain model parameters, after encoding each frame, estimating the frame budget B for the next frame, and mapping the frame budget to a frame QP (e.g., 1 to 31) using budget-to-rho and rho-to-QP mappings, e.g., via either a rho table or a rho parametric equation.

[00107] Upon post-processing of the QP values, including any additional constraints on frame QP, the embedded logic core sends the frame QP, rho budget and new model parameters to the DSP. The DSP then calculates the QP for each MB using the rho-to-QP mapping, and performs post-processing of the QP values. The DSP may preserve a rule that the MB delta QP value is within +2 and -2, as well as any additional constraints on MB QPs. Upon updating the rho domain model parameters after encoding a MB, the DSP repeats the process for the other MBs within the applicable video frame. After MB encoding is completed, the process returns to the embedded logic core to handle the next video frame to be encoded.

[00108] Video encoder system 12 may have a dedicated memory for storing instructions and data, as well as dedicated hardware, software, firmware, or combinations thereof. If implemented in software, the techniques may be embodied as instructions on a computer-readable medium such as random access memory (RAM), read-only memory (ROM), non-volatile random access memory (NVRAM), electrically erasable programmable read-only memory (EEPROM), FLASH memory, or the like. The instructions cause one or more processors to perform certain aspects of the functionality described in this disclosure.

[00109] The techniques described in this disclosure may be implemented within a general purpose microprocessor, digital signal processor (DSP), application specific integrated circuit (ASIC), field programmable gate array (FPGA), or other equivalent logic devices. For example, video encoder system 12, video decoder system 14, and associated components and modules, may be implemented as parts of an encoding process, or coding/decoding (CODEC) process, running on a digital signal processor (DSP) or other processing device. Accordingly, components described as modules may form programmable features of such a process, or a separate process.

[00110] Video encoder system 12 may have a dedicated memory for storing instructions and data, as well as dedicated hardware, software, firmware, or combinations thereof. If implemented in software, the techniques may be embodied as instructions executable by one or more processors. The instructions may be stored on a computer-readable medium such as random access memory (RAM), read-only memory (ROM), non-volatile random access memory (NVRAM), electrically erasable programmable read-only memory (EEPROM), FLASH memory, magnetic or optical data storage device, or the like. The instructions cause one or more processors to perform certain aspects of the functionality described in this disclosure.

[00111] Various embodiments have been described. These and other embodiments are within the scope of the following claims.

CLAIMS

1. A method comprising:
 - estimating video throughput based on a size of a video flow radio link protocol (RLP) queue at an access terminal; and
 - encoding video data using the estimated video throughput.

2. A method comprising:
 - determining a first size V_n of a video queue in a radio link protocol (RLP) layer at a first time t_n based on a video frame rate;
 - determining a second size V_m of the video queue at a second time t_m based on an audio frame rate;
 - if the first size V_n or the second size V_m is greater than zero, then using the first size V_n , a previous size V_{n-1} of the video queue associated with a previous video frame, a previous video frame size B_{n-1} , the first time t_n , and a time t_{n-1} associated with the previous size of the video queue to determine a video throughput VTP;
 - if the first size V_n and the second size V_m are equal to zero, then searching for an earlier time based on the audio frame rate when the video queue size was greater than zero;
 - after finding the earlier time based on the audio frame rate when the video queue size was greater than zero, using an earlier queue size V_{m-i} based on the audio frame rate, the previous size V_{n-1} of the video queue associated with the previous video frame, the previous video frame size B_{n-1} , the earlier time t_{m-1} , and the time t_{n-1} associated with the previous size of the video queue to determine a video throughput VTP;
 - using the determined video throughput VTP to determine a channel-constrained video frame size; and
 - using the channel-constrained video frame size to control a video encoding rate.

3. The method of Claim 2, wherein using the first size (V_x), a previous size (V_{n-1}) of the video queue associated with a previous video frame, a previous video frame size (B_{n-1}), the first time (t_x), and a time (t_{n-1}) associated with the previous size of the video queue to determine a video throughput (VTP) comprises:

$$VTP = \frac{B_{n-1} + V_{n-1} - V_x}{t_x - t_{n-1}},$$

4. The method of Claim 2, wherein using the earlier size (V_x), the previous size (V_{n-1}) of the video queue associated with the previous video frame, the previous video frame size (B_{n-1}), the earlier time (t_x), and the time (t_{n-1}) associated with the previous size of the video queue to determine a video throughput (VTP) comprises:

$$VTP = \frac{B_{n-1} + V_{n-1} - V_x}{t_x - t_{n-1}},$$

5. The method of Claim 2, further comprising if the first size and the second size are equal to zero, and a third size of the video queue associated with a previous audio frame is greater than zero, then using the third size, the previous size of the video queue associated with a previous video frame, the previous video frame size, a time associated with the third size, and the time associated with the previous size of the video queue to determine a video throughput.

6. The method of Claim 2, wherein the video queue is assigned to only store video packets.

7. The method of Claim 2, wherein using the determined video throughput to determine a channel-constrained video frame size further uses the video frame rate, the first size and an adjustment factor, which controls how much the video encoding rate reacts to channel conditions.

8. The method of Claim 7, wherein the adjustment factor has a first value for video conferencing and a second value for video sharing.

9. The method of Claim 2, further comprising:
 using the video encoding rate to encode video; and
 transmitting the encoded video across a wireless channel using Code Division Multiple Access Evolution Data Optimized Revision A (CDMA EV-DO Rev A) protocols.

10. The method of Claim 2, further comprising:
 determining whether the channel-constrained video frame size is greater than a minimum frame size threshold;

if the channel-constrained video frame size is greater than the minimum frame size threshold, determining whether the channel-constrained video frame size is smaller than a frame size constrained by a virtual buffer;

if the channel-constrained video frame size is smaller than the frame size constrained by the virtual buffer, then using the channel-constrained video frame size to control the video encoding rate; and

if the channel-constrained video frame size is greater than the frame size constrained by the virtual buffer, then using the frame size constrained by the virtual buffer to control the video encoding rate.

11. A method comprising:

determining a size of a video queue in a radio link protocol (RLP) layer;

determining a power headroom limitation from a medium access control (MAC) layer;

using the determined power headroom limitation to determine a MAC payload size;

using the determined MAC payload size and an estimate of how many transmission opportunities are given to video in a time period to determine video throughput;

using the determined video throughput and the determined size of the video queue in the RLP layer to determine a channel-constrained video frame size; and

using the channel-constrained video frame size to control a video encoding rate.

12. The method of Claim 11, wherein using the determined video throughput and the determined size of the video queue in the RLP layer to determine a channel-constrained video frame size further uses an adjustment factor, which controls how much the video encoding rate reacts to channel conditions.

13. The method of Claim 12, wherein the adjustment factor has a first value for video conferencing and a second value for video sharing.

14. The method of Claim 11, further comprising:

determining whether the channel-constrained video frame size is greater than a minimum frame size threshold;

if the channel-constrained video frame size is greater than the minimum frame size threshold, determining whether the channel-constrained video frame size is smaller than a frame size constrained by a virtual buffer;

if the channel-constrained video frame size is smaller than the frame size constrained by the virtual buffer, then using the channel-constrained video frame size to control the video encoding rate; and

if the channel-constrained video frame size is greater than the frame size constrained by the virtual buffer, then using the frame size constrained by the virtual buffer to control the video encoding rate.

15. The method of Claim 11, further comprising:

using the video encoding rate to encode video; and

transmitting the encoded video across a wireless channel using Code Division Multiple Access Evolution Data Optimized Revision A (CDMA EV-DO Rev A) protocols.

16. The method of Claim 11, wherein the video queue is assigned to only store video packets.

17. An apparatus comprising a machine-readable memory storing a set of instructions configured to:

determine a first size V_n of a video queue in a radio link protocol (RLP) layer at a first time t_n based on a video frame rate;

determine a second size V_m of the video queue at a second time t_m based on an audio frame rate;

if the first size V_n or the second size V_m is greater than zero, then use the first size V_n , a previous size V_{n-1} of the video queue associated with a previous video frame, a previous video frame size B_{n-1} , the first time t_n , and a time t_{n-1} associated with the previous size of the video queue to determine a video throughput VTP;

if the first size V_n and the second size V_m are equal to zero, then searching for an earlier time based on the audio frame rate when the video queue size was greater than zero;

after finding the earlier time based on the audio frame rate when the video queue size was greater than zero, use an earlier queue size V_{m-i} based on the audio frame rate, the previous size V_{n-1} of the video queue associated with the previous video frame, the previous video frame size B_{n-1} , the earlier time t_{m-1} , and the time t_{n-1} associated with the previous size of the video queue to determine a video throughput VTP;

use the determined video throughput VTP to determine a channel-constrained video frame size; and

use the channel-constrained video frame size to control a video encoding rate.

18. The apparatus of Claim 17, wherein instructions operate on an application layer above the RLP layer.

19. The apparatus of Claim 17, wherein a medium access control (MAC) layer for video flows of the apparatus does not react to sector loading.

20. The apparatus of Claim 17, wherein using the first size (V_x), a previous size (V_{n-1}) of the video queue associated with a previous video frame, a previous video frame size (B_{n-1}), the first time (t_x), and a time (t_{n-1}) associated with the previous size of the video queue to determine a video throughput (VTP) comprises:

$$VTP = \frac{B_{n-1} + V_{n-1} - V_x}{t_x - t_{n-1}},$$

21. The apparatus of Claim 17, wherein using the earlier size (V_x), the previous size (V_{n-1}) of the video queue associated with the previous video frame, the previous video frame size (B_{n-1}), the earlier time (t_x), and the time (t_{n-1}) associated with the previous size of the video queue to determine a video throughput (VTP) comprises:

$$VTP = \frac{B_{n-1} + V_{n-1} - V_x}{t_x - t_{n-1}},$$

22. The apparatus of Claim 17, wherein the instructions further comprise if the first size and the second size are equal to zero, and a third size of the video queue associated

with a previous audio frame is greater than zero, then using the third size, the previous size of the video queue associated with a previous video frame, the previous video frame size, a time associated with the third size, and the time associated with the previous size of the video queue to determine a video throughput.

23. The apparatus of Claim 17, wherein the video queue is assigned to only store video packets.

24. The apparatus of Claim 17, wherein using the determined video throughput to determine a channel-constrained video frame size further uses the video frame rate, the first size and an adjustment factor, which controls how much the video encoding rate reacts to channel conditions.

25. The apparatus of Claim 24, wherein the adjustment factor has a first value for video conferencing and a second value for video sharing.

26. The apparatus of Claim 17, further comprising:
a video encoder configured to use the video encoding rate to encode video; and
a transmitter configured to transmit the encoded video across a wireless channel using Code Division Multiple Access Evolution Data Optimized Revision A (CDMA EV-DO Rev A) protocols.

27. The apparatus of Claim 17, wherein the instructions further comprise:
determining whether the channel-constrained video frame size is greater than a minimum frame size threshold;
if the channel-constrained video frame size is greater than the minimum frame size threshold, determining whether the channel-constrained video frame size is smaller than a frame size constrained by a virtual buffer;
if the channel-constrained video frame size is smaller than the frame size constrained by the virtual buffer, then using the channel-constrained video frame size to control the video encoding rate; and

if the channel-constrained video frame size is greater than the frame size constrained by the virtual buffer, then using the frame size constrained by the virtual buffer to control the video encoding rate.

28. An apparatus comprising a machine-readable memory storing a set of instructions configured to:

determine a size of a video queue in a radio link protocol (RLP) layer;

determine a power headroom limitation from a medium access control (MAC) layer;

use the determined power headroom limitation to determine a MAC payload size;

use the determined MAC payload size and an estimate of how many transmission opportunities are given to video in a time period to determine video throughput;

use the determined video throughput and the determined size of the video queue in the RLP layer to determine a channel-constrained video frame size; and

use the channel-constrained video frame size to control a video encoding rate.

29. The apparatus of Claim 28, wherein using the determined video throughput and the determined size of the video queue in the RLP layer to determine a channel-constrained video frame size further uses an adjustment factor, which controls how much the video encoding rate reacts to channel conditions.

30. The apparatus of Claim 28, wherein the instructions further comprise:

determining whether the channel-constrained video frame size is greater than a minimum frame size threshold;

if the channel-constrained video frame size is greater than the minimum frame size threshold, determining whether the channel-constrained video frame size is smaller than a frame size constrained by a virtual buffer;

if the channel-constrained video frame size is smaller than the frame size constrained by the virtual buffer, then using the channel-constrained video frame size to control the video encoding rate; and

if the channel-constrained video frame size is greater than the frame size constrained by the virtual buffer, then using the frame size constrained by the virtual buffer to control the video encoding rate.

31. An apparatus comprising:

a radio link protocol (RLP) layer queue configured to store video data;

a first unit configured to receive a size of the RLP video queue and a power headroom limitation from a medium access control (MAC) layer, use the power headroom limitation to determine a MAC payload size, use the determined MAC payload size and an estimate of how many transmission opportunities are given to video in a time period to determine video throughput, and use the determined video throughput and the size of the video queue in the RLP layer to determine a channel-constrained video frame size;

a second unit to use the channel-constrained video frame size to control a video encoding rate; and

a video encoder to use the video encoding rate to encode video.

32. The apparatus of Claim 31, further comprising:

a virtual buffer; and

a third unit configured to:

determine whether the channel-constrained video frame size is greater than a minimum frame size threshold;

if the channel-constrained video frame size is greater than the minimum frame size threshold, determine whether the channel-constrained video frame size is smaller than a frame size constrained by a virtual buffer;

if the channel-constrained video frame size is smaller than the frame size constrained by the virtual buffer, then use the channel-constrained video frame size to control the video encoding rate; and

if the channel-constrained video frame size is greater than the frame size constrained by the virtual buffer, then use the frame size constrained by the virtual buffer to control the video encoding rate.

33. An apparatus comprising:
- means to determine a size of a video queue in a radio link protocol (RLP) layer;
 - means to determine a power headroom limitation from a medium access control (MAC) layer;
 - means to use the determined power headroom limitation to determine a MAC payload size;
 - means to use the determined MAC payload size and an estimate of how many transmission opportunities are given to video in a time period to determine video throughput;
 - means to use the determined video throughput and the determined size of the video queue in the RLP layer to determine a channel-constrained video frame size; and
 - means to use the channel-constrained video frame size to control a video encoding rate.