Application-Driven Cross-Layer Optimization for Video Streaming over Wireless Networks

S. Khan, Y. Peng, and E. Steinbach, Technische Universität München
M. Sgroi and W. Kellerer, Docomo Communications Laboratories Europe GmbH

ABSTRACT

Mobile multimedia applications require networks that optimally allocate resources and adapt to dynamically changing environments. Cross-layer design (CLD) is a new paradigm that addresses this challenge by optimizing communication network architectures across traditional layer boundaries. In this article we discuss the relevant technical challenges of CLD and focus on application-driven CLD for video streaming over wireless networks. We propose a cross-layer optimization strategy that jointly optimizes the application layer, data link layer, and physical layer of the protocol stack using an application-oriented objective function in order to maximize user satisfaction. In our experiments we demonstrate the performance gain achievable with this approach. We also explore the trade-off between performance gain and additional computation and communication cost introduced by cross-layer optimization. Finally, we outline future research challenges in CLD.

INTRODUCTION

Media streaming, video conferencing, and interactive networked 3D games are examples of applications that are expected to attract an increasing number of users in the future. Supporting these applications will be a major challenge for beyond third-generation (B3G) wireless networks. Existing mobile multimedia systems are specifically tailored to individual wireless network technologies and do not address the heterogeneity that is expected in B3G networks. Mobile users expect high quality and transparent service independent of the network access technology. At the same time, network operators need to efficiently allocate the wireless resources to increase network capacity and provide services to as many users as possible at the best possible quality level.

The traditional approach to network design is to identify a stack of layers and design them in isolation. The layered approach has been widely used in the past, but it is no longer adequate to meet the challenges of next-generation mobile systems. Mobile multimedia communication is especially challenging due to the time varying transmission characteristics of the wireless channel and the dynamic quality of service (QoS) requirements of the application (e.g., variable bit rate, prioritized delivery of important media units, and variable tolerance vs. bit or packet errors). Setting the control modes and tuning the parameters of the protocols at design time and for the worst case scenarios lead to poor performance and inefficient utilization of resources. Instead, a network observing the behavior of the application and of the physical channel and dynamically adapting to the changes is able to maintain optimal allocation of resources. This requires timely exchange of parameters across layers and periodic reconfiguration of modes and parameters of the protocol layers during network operation.

Cross-layer design (CLD) is a new paradigm in network architecture design that takes into account the dependencies and interactions among layers, and supports optimization across layer boundaries [1, 2]. A common misconception about CLD is that it consists of designing networks without layers. CLD should not be viewed as an alternative to the layered approach, but rather as a *complement*. Layering and cross-layer optimization are tools that should be used together to design highly adaptive wireless networks.

Previous work on CLD has mainly focused on optimizing the performance of a single layer, for example, adapting the application to the transport, network, data link, and physical layer characteristics (bottom-up approach), or adapting the physical, data link, or network layers to the application requirements (top-down approach). The bottom-up approach typically exploits information about the current channel situation to adapt the transmission policy of the application [3]. Recent approaches to channel-adaptive video streaming are described in [4]. The top-down approach typically passes priority labels to the lower layers, which perform, for

0163-6804/06/$20.00 © 2006 IEEE

instance, class-based queuing and priority-based transmission.

Most of the ongoing research in CLD focuses on joint optimization of the physical layer and data link (or MAC) layer [5]. Only recently have approaches that explicitly include the application in cross-layer optimization appeared [6–13]. For instance, [6] schedules packet transmission over orthogonal frequency-division multiplexing (OFDM) channels giving higher priority to the most important packets (I-frames). The approach in [6] takes the size of the queues into account in order to determine the number of OFDM subcarriers to allocate to each user. Also, the observed quality of the channels is explored to assign to each terminal a set of sub-carriers in a good transmission state. The authors in [7] define an opportunistic scheduling algorithm for multiple video streams using a priority function that depends on channel conditions, importance of frames, queue size, and multiplexing gain. The authors in [8] propose a cross-layer scheduling framework with adaptive modulation and coding. In [9] application layer adaptation mechanisms are combined with lower-layer adaptation strategies for low-delay wireless video streaming. Corrupted data is passed across the layer boundaries, and retransmissions and forward error correction (FEC) are performed end to end at the application layer. In [10] different error control and adaptation mechanisms available in the different layers for error-robust video transmission are evaluated. The mechanisms considered include retransmission on the data link layer, application-layer FEC, scalable video coding, and adaptive packetization. The authors in [11] propose a new paradigm for wireless communications based on *coopetition*, which allows wireless stations to optimally and dynamically adapt their cross-layer transmission strategies to improve multimedia quality and power consumption. In [12, 13] an application-driven cross-layer optimization framework for wireless video streaming is introduced that forms the basis of this article.

In this article we present a cross-layer optimization approach to wireless video streaming where layer-specific information is passed in both directions, top-down and bottom-up. We consider joint optimization of three layers of the protocol stack: the application, data link, and physical layers. The application layer contributes to joint optimization because it has knowledge of the distortion effect of each packet loss on user-perceived quality and can dynamically adapt the source rate and media transmission strategy to current network capabilities. The physical and data link layers are also taken into consideration because they estimate the transmission capabilities of the wireless medium and quickly adapt to its variations. We regard our approach as an application-driven cross-layer optimization, as our main concern is to maximize user satisfaction using an application-specific objective function.

Our work is novel in several aspects. First, we jointly optimize parameters of multiple layers taking into consideration effective abstractions of the application, data link, and physical layers.

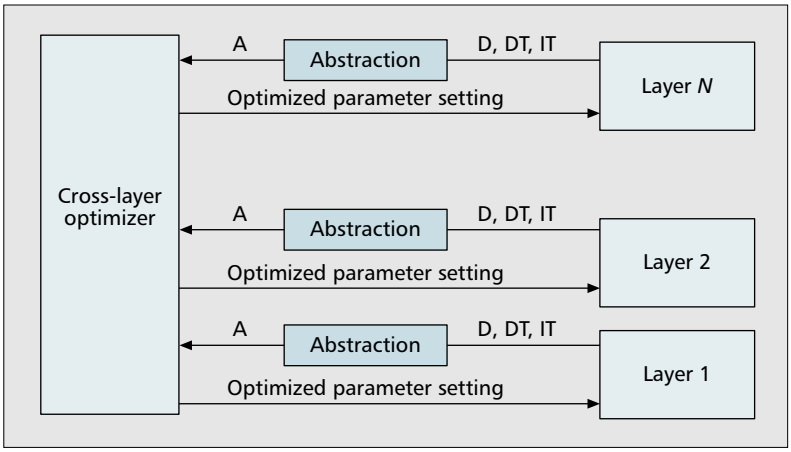

Figure 1. *Cross-layer architecture.*

Also, the proposed cross-layer optimizer uses an application-based objective function. Second, not only do we evaluate the performance gain of cross-layer optimization through experiments on a testbed; we also discuss the trade-off between the performance gain and the additional computation and communication cost of the optimization.

Cross-Layer Architecture

Our cross-layer architecture (CLA) is composed of N layers and a cross-layer optimizer (CLO), as visualized in Fig. 1. The CLO jointly optimizes multiple network layers, making predictions on their states and selecting optimal values for their parameters.

The proposed cross-layer optimization concept consists of three steps:

1) Layer abstraction computes an abstraction of layer-specific parameters. The number of parameters used by the CLO is significantly reduced by the abstraction process.

2) Optimization finds the values of layer parameters that optimize a specific objective function. In our work, the objective function is a function of the expected video reconstruction quality of multiple users.

3) Layer reconfiguration distributes the optimal values of the abstracted parameters to the corresponding layers. It is the responsibility of the individual layers to translate the selected abstracted parameters back into layer-specific parameters and actual modes of operation.

These steps are repeated at a rate that depends on how fast the application requirements and transmission capabilities of the physical medium vary. Identifying the parameters that describe the capabilities of a layer is a critical step. A layer description with a large set of parameters is accurate but usually results in high cost in terms of data processing and communication overhead. Therefore, abstractions have to be used to reduce the number of parameters. Also, abstracted parameters hide the actual technology and therefore allow us to design the cross-layer optimizer in a more general way and to use the same optimizer in different systems. From a system perspective, there are different

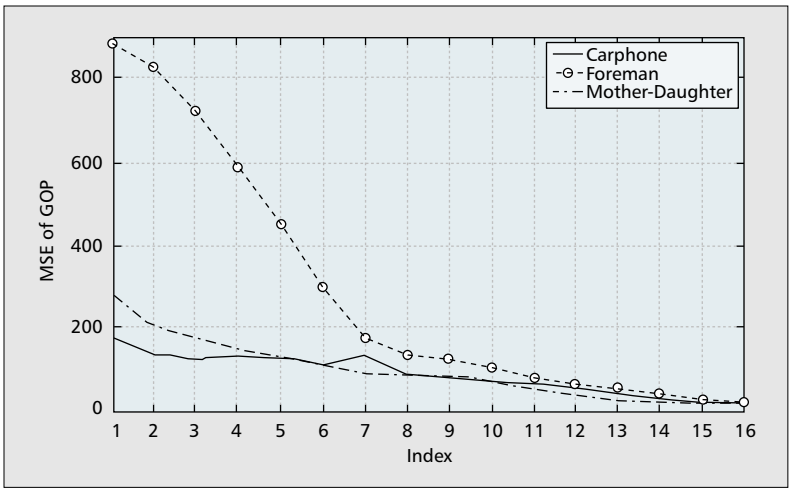

Figure 2. *Distortion profile. The index on the horizontal axis expresses the first frame lost in the GOP. The vertical axis shows the resulting reconstruction distortion assuming that all depending frames are concealed using copy-previous-frame error concealment.*

kinds of parameters involved, which can be classified as follows:

Directly tunable (DT) parameters: These can be set directly as a result of the CLO. Examples: time slot assignment in a time-division multiple access (TDMA) system or carrier assignment in an OFDM system.

Indirectly tunable (IT) parameters: These cannot be set directly as a result of the CLO, but may change as a result of the setting of DT parameters. Example: bit error rate that depends on the type of coding and modulation scheme adopted.

Descriptive (D) parameters: These can be read by the CLO, but cannot be tuned. Examples: frame rate or picture size in streaming video applications that are set at encoding time, channel quality estimates obtained from channel estimation.

Abstracted (A) parameters: These are abstractions of descriptive, DT, and IT parameters used in the CLO. Example: net transmission rate and transition probabilities of a two-state packet erasure model (Gilbert-Elliot model), for instance, as used in [12].

Cross-Layer Optimization for Wireless Video Streaming

Videos have variable data rates due to the dynamic nature of the captured scene and encoding. A video stream is usually encoded as a sequence of groups of consecutive frames, called a group of pictures (GOP). In a GOP the first frame (I-frame) is encoded independent of other frames, and the remaining frames (P-frames, B-frames) are differentially encoded with respect to other (mostly previous) frames in the same GOP. A frame can be successfully decoded at the receiver if all frames in the GOP on which it depends have been correctly received and decoded. Therefore, the amount of distortion at the receiver varies depending on which frame is lost. For example, losing an I-frame results in higher distortion than losing a P frame, as shown in Fig. 2 that visualizes the mean square error (MSE) distortion profile of a GOP (including 15 frames: one I-frame and 14 P-frames) measured for three test sequences encoded in H.264/AVC format.

The distortion profiles shown in Fig. 2 have been determined as follows. One GOP (15 frames) of a test sequence is encoded using the H.264/AVC video compression standard. The decoding is performed assuming that a particular frame (called index in Fig. 2) is the first frame in the GOP that is not received correctly. Copy-previous-frame error concealment is performed for all following frames in the same GOP. Index 1 in Fig. 2 represents the case that the first frame of the GOP (I-frame) is lost and concealment is performed using the most recent correctly displayed frame from the previous GOP. Similarly, index 2 means that the first P-frame is the first frame lost in this GOP. In this case the lost frame and all following frames are replaced by the decoded I-frame. Index 15 refers to the case where the final P-frame in the GOP is the first frame lost, which means that only the last frame in the GOP is concealed by displaying the second-to-last frame. Index 16 represents the case where all frames are received correctly, and hence no error concealment is employed. The later in the GOP the first frame loss, the smaller the reconstruction error. Figure 2 also shows that the actual distortion heavily depends on the characteristics of the video sequence. While the loss of a particular frame in one video sequence may have only little influence on the reconstruction quality, losing the same frame in a different video may lead to a dramatic reduction in quality. This leads to opportunities for dynamic resource allocation across multiple users, which is one of the key components of our proposed CLD concept.

The main challenge of wireless video streaming is to ensure timely delivery of video frames at the client despite changing channel conditions, ensuring reasonable perceptual quality for the user. A truly efficient use of network resources and optimization of end-to-end quality in mobile networks requires adaptability to changing application and network characteristics on all layers. The application layer can adapt to varying network characteristics by adequate processing, for example, dynamic rate adaptation at the video server or joint adaptation of application source rate and application layer code rate (joint source channel coding). Adaptation can also take place at the lower layers (e.g., by using adaptive modulation and coding, adaptive beamforming and scheduling). For an overview of properties and challenges of video streaming the reader is referred to [4].

Applying cross-layer optimization to multiple layers, including the application, network, data link, and physical layers that directly interface with the dynamically changing environment, allows for optimal adaptation of the network. For example, joint optimization using predictions of the states of the channels for all users, together with knowledge of the type of frame

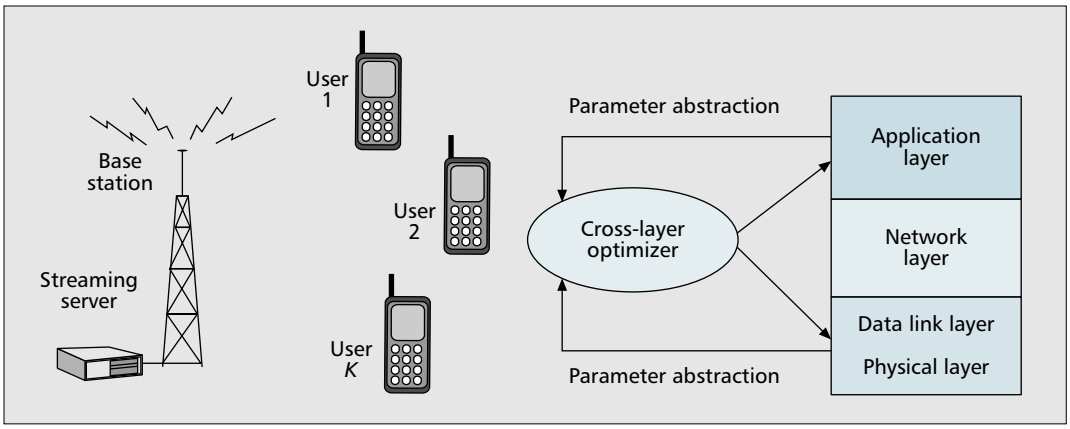

Figure 3. *Application scenario (left) and cross-layer optimization concept (right) for wireless video streaming.*

The objective of the optimization is to maximize the video quality perceived by the users. As a measure of the user-perceived video quality we use the expected PSNR at the receiver. The objective function can be defined in different ways, for example as the video quality of individual users or the average video quality among all the users.

carried by each packet favors at any time transmission of the most important frames over channels with the best transmission capabilities. When multiple users share the same wireless resources (e.g., they are located in the same cell), a multi-user diversity gain can be achieved by assigning, whenever possible, the channels to users who have high probability of successful transmission [1].

Examples of parameters that can be jointly optimized are:

- Source rate, encoding format, compression, FEC (application layer)
- Route (network layer)
- TDMA time slots, OFDM carriers, directional beams, FEC (data link layer)
- Modulation scheme, channel coding, power (physical layer)

In order to formulate an application-oriented cross-layer optimization concept, the application layer quality has to be quantified appropriately. For video streaming, the so-called peak signal-to-noise ratio (PSNR) is a quantitative parameter that closely represents user-perceived video quality and therefore can be used to define an objective function for optimization of video streaming delivery systems. In the following we describe the cross-layer optimized wireless video streaming scenario employed in this work.

Video Streaming Scenario and Architecture

Let us consider an application scenario where a base station delivers streaming videos to K mobile users located in its cell (e.g., three users on the left side of Fig. 3). The right side of Fig. 3 shows our cross-layer optimization concept where the application, data link, and physical layers are jointly optimized. Cross-layer optimization is applied to all the users simultaneously in order to optimally allocate resources and take advantage of multi-user diversity.

The system is optimized periodically at the beginning of each GOP. First, the CLO takes an abstraction of the parameters of the different layers. The physical and data link layers are abstracted by the transition probabilities of a two-state Markov packet burst loss (Gilbert-Elliott) model, described next. The application layer is abstracted by the rate distortion profile described later.

In the following example we assume that after the process of abstraction, the CLO optimizes the multi-user system by selecting the optimal values of the following layer-specific parameters:

- The video source rate (application layer)
- The time slot allocation (data link layer)
- The modulation scheme (physical layer)

The objective of the optimization is to maximize the video quality perceived by users. As a measure of user-perceived video quality we use the expected PSNR at the receiver. The objective function can be defined in different ways; for example, as the video quality of individual users (e.g., the user experiencing the worst video quality among all users) or the average video quality among all users.

In our experiments the objective function is chosen to be the average PSNR of all users,

$$F(\tilde{\mathbf{x}}) = \frac{1}{K}\sum_{k=1}^{K} PSNR_k(\tilde{\mathbf{x}}),$$

where $F(\tilde{\mathbf{x}})$ is the objective function with the cross-layer parameter tuple $\tilde{\mathbf{x}} \in \tilde{X}$. $\tilde{X}$ is the set of all possible parameter tuples abstracted from the protocol layers. The decision of the optimizer can be expressed as

$$\tilde{\mathbf{x}}_{opt} = \arg\max_{\tilde{\mathbf{x}}\in\tilde{X}} F(\tilde{\mathbf{x}}),$$

where $\tilde{\mathbf{x}}_{opt}$ is the optimum parameter tuple that maximizes the objective function.

Once the CLO has selected the optimal values of the abstracted parameters, it distributes them to all the individual layers, which are responsible for translating them back into actual modes of operation.

Radio Link Layer Abstraction

To abstract the physical layer we use the Gilbert-Elliot (GE) packet erasure channel model that is known to characterize the fading of a wireless channel with sufficient accuracy. The GE model represents the dynamics of the packet error behavior of a wireless channel with two states, denoted G (good) and B (bad). In state G packets are assumed to be received correctly and in a

As most of the modern video codecs employ a predictive coding structure, encoders produce a variable rate bit stream which is highly susceptible to packet loss. Application-layer abstraction becomes necessary as the optimizer needs to be aware of the effects of lower layer parameters on the application layer.

timely manner, whereas in state B packets are assumed to be lost. This model can be described by the transition probabilities p from state G to B and q from state B to G. For each user, four key parameters are abstracted at the radio link layer:

- Transmission data rate
- Transmission packet error rate
- Data packet size
- Channel coherence time

These four parameters form the abstracted parameter tuple that describes the radio link layer (radio link layer = physical + data link layer). Depending on the actual number of usable operational modes on the radio link layer, the optimizer may potentially receive a large number of candidate tuples from the radio link layer abstraction. The transmission data rate is influenced by the modulation scheme, channel coding, and multi-user scheduling. The transmission packet error rate is influenced by the transmit power, channel estimation, signal detection, modulation scheme, channel coding, and so on. The channel coherence time of a user is related to the user velocity and its surrounding environment, while the data packet size is usually defined by the wireless system standard. We compute the transition probabilities (p and q) of each user with the help of these abstracted parameters, as detailed in [12, 15]. We consider retransmission of the most important frames whenever the transmission rate allocated to a user is larger than the video source rate, which reduces the packet error probability of the retransmitted packets.

Application Layer Abstraction

As most modern video codecs employ a predictive coding structure, encoders produce a variable rate bitstream that is highly susceptible to packet loss. Application layer abstraction becomes necessary as the optimizer needs to be aware of the effects of lower-layer parameters on the application layer. A convenient abstraction tool for streaming video is the *rate distortion profile* introduced in [16]. Side information is sent along with the regular video bitstream. This information consists of a rate vector that shows the size of the video frames in bytes and a distortion matrix with entries that allow us to compute the reconstruction distortion for the displayed GOP for arbitrary loss patterns.

Cross-Layer Optimization

The cross-layer optimizer selects for each GOP the optimal parameter values that maximize the expected user-perceived video quality. This requires computing for each user and each parameter set the expected video reconstruction quality at the receiver, which is obtained as the sum of the source distortion D_S and the expected loss distortion D_L:

$$D = D_S + D_L$$

The source distortion D_S is the reconstruction quality obtained in the error-free case and has to be sent as side information along with the video bitstream. This distortion is a function of source rate. The larger the source rate, the smaller the source distortion. The loss distortion D_L, in contrast, is a function of the packet loss rate observed during transmission. As different loss patterns lead to different reconstruction distortions, the expected value of loss distortion is used in our work. Using the information from the rate vector and distortion matrix in combination with the transition probabilities from the two-state Markov packet burst loss model, the expected loss distortion can be computed as

$$D_L = \sum_{i=1}^{l} p_i \cdot D_i,$$

where l is the number of different loss patterns [12, Fig. 2], p_i is the loss pattern probability, and D_i is the resulting reconstruction distortion for loss pattern i derived from the distortion matrix. The probability of a particular loss pattern p_i is computed from the transition probabilities of the Gilbert-Elliot model as described in [12]. Once the expected user quality is available for every application and radio link layer parameter set, the task of the optimizer simply becomes choosing the best operating point with respect to the desired objective function. Please note that in an earlier section PSNR was used as the quality measure, while here MSE is used for D_S and D_L. PSNR, however, is simply a logarithmic form of D and can be computed as

$$PSNR = 10 \cdot \log_{10}\left(\frac{255^2}{D}\right).$$

Simulation Results

We simulate a wireless video streaming scenario with three users, each requesting a different video from the streaming server located at the base station. The video sequences are Mother and Daughter (MD), Carphone (CP), and Foreman (FM). All the three videos are in QCIF resolution (176 × 144) with a frame rate of 30 frames/s. The videos are pre-encoded at two different target source rates of 100 and 200 kb/s. Each GOP has 15 frames, including one I-frame and 14 P-frames. The average PSNR between the encoded and displayed video sequences is used as a performance measure. Figure 4 illustrates the parameter abstraction from the application and radio link layer for a particular optimization cycle, and illustrates the process of cross-layer optimization.

At the radio link layer, the two-state GE channel model is assumed, with parameters (transition probabilities) p and q. The total transmission capacity of the system is assumed to be 300 ksymbols/s. Two different modulation schemes, binary phase shift keying (BPSK) and quaternary PSK (QPSK) are considered, giving a total rate of 300 and 600 kb/s, respectively. Each user has a set of possible transmission rates of {0,100,150,200,300} kb/s. If the available transmission rate exceeds the source rate, the most important frames of the GOP are repeatedly transmitted. Taking the total rate constraint and set of transmission rates into account, we have 72 possible rate allocations

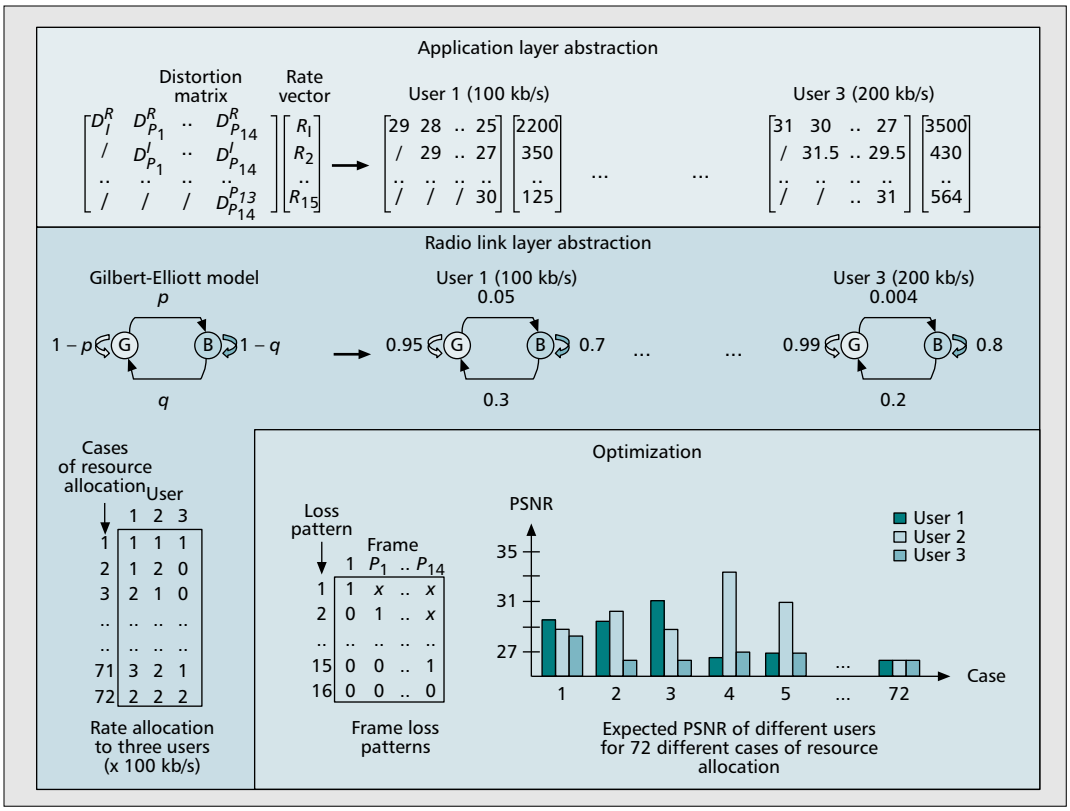

Figure 4. *Cross-layer optimization for a three-user video streaming scenario. The top shows an example of distortion and rate side information sent along with the bitstream; they are used as the parameter abstraction of the application layer. The middle and lower left show the radio link layer abstraction, which consists of the transition probabilities of the two-state Markov model and multi-user resource allocation. The lower right shows the possible loss patterns from which the expected reconstruction distortion is computed. The resulting PSNR values for the three users are shown for the 72 candidate application and radio link layer parameter sets. The optimization picks the parameter set that maximizes the objective function of the cross-layer optimizer.*

The cross-layer optimizer selects for each GOP the optimal parameter values that maximize the user-perceived video quality as described earlier. In case the rate-distortion side information is not available, the expected loss distortion can be approximated by computing the ENDEF, which does not use rate distortion side information

among users, as illustrated in the lower left of Fig. 4. With 15 frames in each GOP, we have 16 different loss patterns to consider (Fig. 4). Pattern 1 represents the case when the I-frame is not decodable. Because of frame dependency, all the subsequent frames in the current GOP become nondecodable and are replaced by the last decoded frame in the previous GOP. Pattern 2 represents the case where all the packets in the I-frame are received correctly but at least one packet in frame P1 is lost. The rest may be deduced by analogy. Pattern 16 represents the case without any packet loss. The probability of these patterns can be computed using the parameters of the GE model. We then compute the resulting reconstruction distortion for each loss pattern from the distortion matrix and form the expected value of the reconstruction distortion for every user, as explained earlier.

The cross-layer optimizer selects for each GOP the optimal parameter values that maximize user-perceived video quality as described earlier. If the rate distortion side information is not available, the expected loss distortion can be approximated by computing the expected number of decodable frames (ENDEF), which does not use rate distortion side information and therefore provides a less accurate approximation of the expected PSNR. Working without rate distortion side information reduces the transmission overhead but leads to lower average video quality at the receiver side, as shown in the next section.

Performance Analysis

In this section we compare the performance gain obtained by applying cross-layer optimization for the two cases where expected PSNR (CLO with RD) and ENDEF (CLO without RD) are used by the optimizer to predict video quality. We analyze three different scenarios. Figure 5 shows the cumulative density function (CDF) of the average PSNR for all three scenarios, each based on 1000 sessions. In the first scenario all users have very bad channel conditions, which is reflected in our simulation by randomly varying the receive SNR for each user between 0 and 5 dB. Average PSNR increases about 2 dB for cross-layer optimization with rate distortion side information (CLO with RD) over that without optimization (without CLO). In the second scenario simulations are performed with random user SNR over a large range (0–25 dB) for all users. The curve representing CLO performed without RD side information lies approximately halfway between the other two curves for both scenarios 1 and 2. In

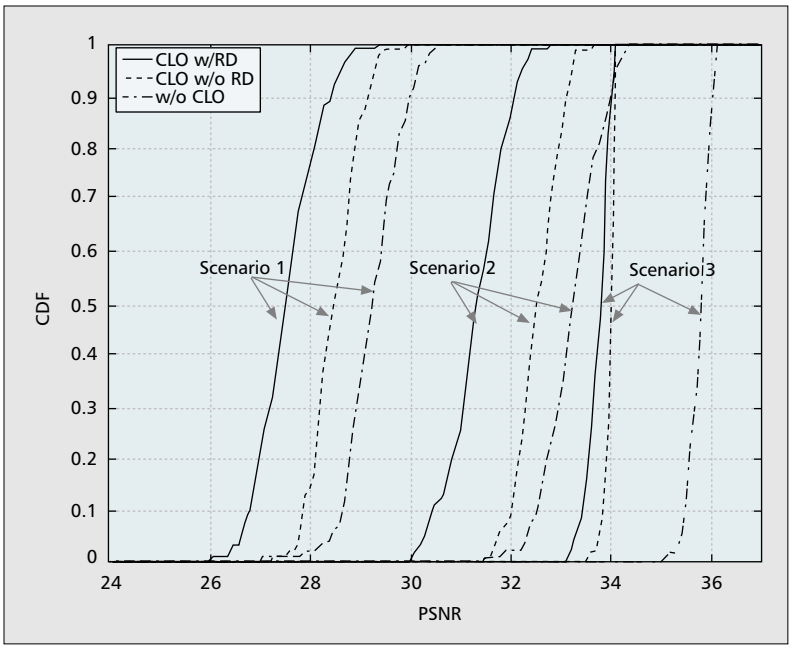

Figure 5. *CDF of average PSNR for random receive SNR from 0 to 5 dB (scenario 1), 0 to 25 dB (scenario 2), and 20 to 25 dB (scenario 3).*

the third scenario all users have very good channel conditions, and the receive SNR varies randomly in the range 20–25 dB. Again, we observe an average PSNR improvement of about 2 dB for cross-layer optimization with RD side information over the case without optimization. In the third scenario the optimizer can take advantage of the good channel conditions by choosing the higher-source-rate (200 kb/s) videos. Performance without RD side information is worse in this case because of the lower correlation between the number of decodable frames and the resulting PSNR.

Cost Analysis

Previous work on CLD has succeeded in showing the potential performance gain over traditional layered communication. However, the additional cost to be paid to perform optimization, and gather the relevant parameters from multiple layers and network locations has typically been neglected.

A CLA has three types of cost in addition to a purely layered one. First, exploring a broad parameter domain space and evaluating the objective function for a large set of value assignments require substantial computation and may result in additional delay. Second, gathering all the parameter abstractions that describe the state and capabilities of different layers may have a nonnegligible communication overhead. Third, network architectures with cross-layer optimizations are less modular and therefore more difficult to manage or reconfigure [2]. Below we measure the first two types of cost in our CLA. The third type is not easy to quantify. However, it can be reduced by defining proper interfaces between the layers and the CLO.

Our cross-layer optimization concept requires computation of the objective function for all candidate parameter tuples. The number of candidate parameter sets increases as a function of the number of users and the number of degrees of freedom in the application and radio link layer available for every user. For example, in Fig. 4 we use 72 candidate resource allocations for three users. For some resource allocations we have the additional choice for some users to either pick a low-rate stream and transmit some frames more than once or pick a high-rate stream with no or fewer frame repetitions. In general, the number of parameter sets to evaluate is the product of the number of possible parameter settings of every degree of freedom in every layer for every user multiplied by the number of users. This number increases rapidly if we consider all possible parameter combinations for a large number of users. In order to keep the number of candidate parameter tuples reasonable, only those abstracted parameter tuples should be evaluated that represent an actual layer-specific operation mode. Not all combinations are feasible due to the limited resources available. In our example in Fig. 4 we restrict the total symbol rate for all three users together to be 300 ksymbols/s. Similarly, a constraint of the total transmit power would typically apply. These constraints reduce the number of possible parameter combinations significantly. Also, different layer-specific operation modes may lead to the same set of abstracted parameters. In this case the layers themselves have to decide which operation mode is more suitable to implement. To summarize, the main challenge of a cross-layer concept such as that described in this article is to keep the number of parameter sets to be evaluated small while preserving the degrees of freedom in the optimization. Recently, this issue has started to be addressed (e.g., [17, 18]).

Although the number of candidate parameter sets increases very rapidly with the number of users, the time to evaluate the different cases increases much slower in our scheme. The computationally most expensive part of the optimization is the computation of the expected reconstruction quality for every user. For a single user, the number of different parameter settings is typically small as long as the number of abstracted parameters per user is small. The computational cost of the remaining task, which involves computing and comparing the objective function for the combination of the parameter settings of all users can be neglected for a small number of users. For a large number of users, however, this becomes increasingly important, as the number of operation modes increases exponentially with the number of users.

If joint optimization of all users becomes too demanding, a simple remedy is to optimize subsets of users. By properly partitioning users into small subsets, complexity can be bounded. The most extreme case would be to form pairs of users that are jointly optimized. While this approach limits the computational load of optimization, the penalty is that the number of degrees of freedom fed to the optimization is significantly reduced, and only a suboptimal solution can be expected.

Transmission of abstracted parameters

across the network causes communication overhead for CLO. In our scheme there is overhead due to transmitting the RD side information from the video server to the cross-layer optimizer. Figure 6 shows the overhead of transmitting the distortion matrix and rate vector for different source rates, assuming that each GOP has one I frame followed only by P frames. It shows that the overhead is quite low, but increases linearly with the number of frames in a GOP.

Optimization using the rate distortion profile provides higher gain due to more accurate calculation of the expected video quality. However, the distortion profile must be transmitted from the server. Moreover, it is not available in applications that require real-time encoding. Our analysis shows that using the expected number of decodable frames still offers a valid gain with respect to the case without CLO, especially for channels with low SNR.

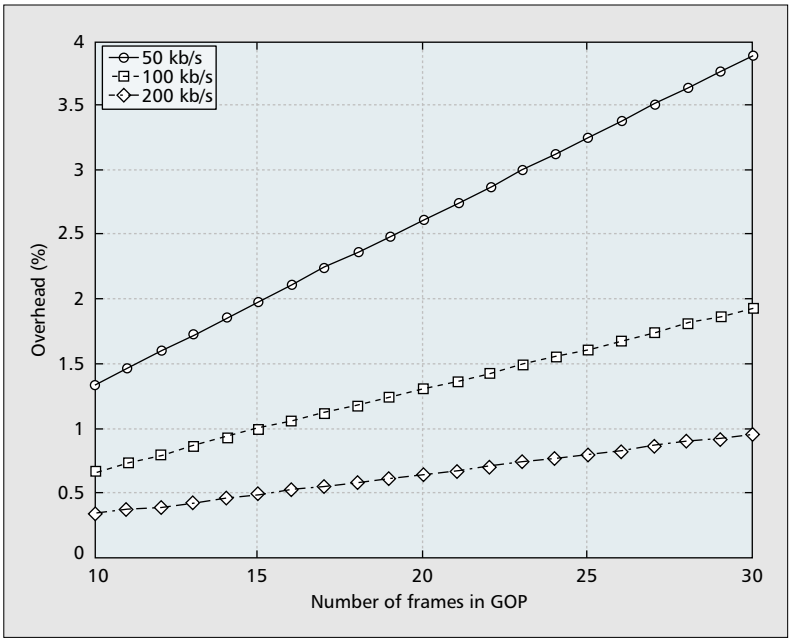

Figure 6. *Traffic overhead of sending RD information as a function of GOP size. The rate vector consists of the frame size in bytes of every frame in the GOP. The number of elements of the rate vector increases linearly as a function of GOP size. The distortion matrix contains all information required to compute the expected reconstruction distortion for arbitrary loss patterns. The number of entries in the distortion matrix increases quadratically with the number of frames in a GOP. Since the size of the bitstream increases linearly with the size of the GOP, the overhead also increases almost linearly. For details on the distortion matrix the interested reader is referred to [16].*

Discussion and Future Work

CLD is a new paradigm that has great potential to change how communication networks are designed and managed in the future. CLD has already been applied successfully in some cases, which have shown that a relevant performance gain can be obtained using a cross-layer architecture instead of a purely layered one. In this work we have introduced an application-oriented cross-layer optimization concept for wireless video streaming. Multi-user resource allocation is performed according to the outcome of the maximization of an objective function that depends on the reconstruction quality on the application layer. Information exchange is performed in both directions, top-down and bottom-up. One of the most important components is the process of parameter abstraction, which helps keep the number of parameter tuples to be evaluated reasonable. Our experimental results show significant quality improvements even for a small number of users and a small number of degrees of freedom within every layer.

However, many technical issues still need to be addressed before when and how to apply CLD is fully understood. The evaluation of the additional cost introduced by cross-layer optimization is one of the most relevant open issues. The additional cross-layer optimization cost has been neglected in most discussions and analysis of CLD, but can be relevant especially in very resource-constrained systems. The decision to apply cross-layer optimization is not an obvious one; therefore, it should be made taking a system approach and trading off all benefits and costs. In particular, a CLD methodology is needed that provides general, yet simple, rules definition on when and how to apply CLD to different types of networks and applications.

An important issue in CLD is definition of the optimization metrics. The goal of CLD is to allocate network resources so that as many users as possible are served with sufficiently high quality. Video quality is highly subjective and therefore difficult to quantify. Furthermore, the metrics of different types of media are hard to compare. Finding a common utility function that allows optimization across concurrent applications of different types would be a useful breakthrough to apply CLD to more realistic and practical cases.

Another issue is the interpretation and role of QoS constraints and fairness. Applying cross-layer optimization improves network capacity and increases the number of users served. However, it is difficult to fully guarantee satisfaction of QoS constraints under all circumstances in a distributed and highly dynamic environment. Temporary shortages of resources may require penalizing the service quality or even interruption of service provisioning for some users. When resources are not sufficient to strictly satisfy all constraints, fairness should be adopted to avoid penalizing the same users.

References

[1] S. Shakkottai, T. Rappaport, and P. Karlsson, "Cross-Layer Design for Wireless Networks," *IEEE Commun. Mag.*, vol. 41, no. 10, Oct. 2003, pp. 74–80.

[2] V. Kawadia and P. Kumar "A Cautionary Perspective on Cross Layer Design," *IEEE Wireless Commun.*, vol. 12, no. 1, Feb. 2005, pp. 3–11.

[3] P. Chou and Z. Miao, "Rate-distortion Optimized Streaming of Packetized Media," *IEEE Trans. Multimedia*, to appear.

[4] B. Girod *et al.*, "Advances in Channel-Adaptive Video Streaming," *IEEE Int'l. Conf. Image Processing*, Rochester, NY, Sept. 22–25, 2002.

[5] T. Holliday and A. Goldsmith, "Optimal Power Control and Source Channel Coding for Delay Constrained Traffic Wireless Channels," *IEEE ICC 2003*, Anchorage, AK, May 11–15, 2003.

Applying cross-layer optimization improves network capacity and increases the number of users being served. However, it is difficult to fully guarantee satisfaction of QoS constraints under all circumstances in a distributed and highly dynamic environment.

[6] J. Gross *et al.*, "Cross-Layer Optimization of OFDM Transmission Systems for MPEG-4 Video Streaming," *Comp. Commun.*, vol. 27, 2004, pp. 1044–55.
[7] R. S. Tupelly, J. Zhang, and E. K. P. Chong, "Opportunistic Scheduling for Streaming Video in Wireless Networks," *37th Annual Conf. Info. Sci. and Sys.*, Baltimore, MD, Mar. 12–14, 2003.
[8] Q. Liu, S. Zhou, and G. Giannakis, "Cross-layer Scheduling with Prescribed QoS Guarantees in Adaptive Wireless Networks," *IEEE JSAC*, vol. 23, no. 5, May 2005, pp. 1056–66.
[9] Y. Shan and A. Zakhor, "Cross Layer Techniques for Adaptive Video Streaming over Wireless Networks," *IEEE Int'l. Conf. Multimedia and Expo*, Lausanne, Switzerland, Aug. 26–29, 2002.
[10] M. van der Schaar et al., "Adaptive Cross-Layer Protection Strategies for Robust Scalable Video Transmission over 802.11 WLANs," *IEEE JSAC*, vol. 21, no. 10, Dec. 2003, pp. 1752–63.
[11] M. van der Schaar and S. Shankar, "Cross-Layer Wireless Multimedia Transmission: Challenges, Principles, and New Paradigms," *IEEE Wireless Commun.*, vol. 12, no. 4, Aug. 2005, pp. 50–58.
[12] Y. Peng et al., "Adaptive Resource Allocation and Frame Scheduling for Wireless Multi-user Video Streaming," *IEEE Int'l. Conf. Image Proc.,* Genova, Italy, Sept. 11–14, 2005.
[13] L.-U Choi, W. Kellerer, and E. Steinbach, "Cross-Layer Optimization for Wireless Multi-user Video Streaming," *IEEE Int'l. Conf. Image Proc.*, Singapore, Oct. 24–27, 2004.
[14] D. Wu *et al.*, "Streaming Video Over the Internet: Approaches and Directions," *IEEE Trans. Circuits and Sys. for Video Tech.*, vol. 11, no. 3, Mar. 2001, pp. 282–300.
[15] M. T. Ivrlac, "Parameter Selection for the Gilbert-Elliott Model," Tech. rep. TUM-LNS-TR-03-05, Inst. for Circuit Theory and Sig. Proc., Munich Univ. of Technology, May 2003.
[16] W. Tu, W. Kellerer, and E. Steinbach, "Rate-Distortion Optimized Video Frame Dropping on Active Network Nodes," *Packet Video Wksp. 2004*, Irvine, CA, Dec. 13–14, 2004.
[17] M. T. Ivrlac and J. A. Nossek, "Cross Layer Optimization — an Equivalence Class Approach," *ITG Wksp. Smart Antennas*, Munich, Germany, Mar. 18–19, 2004.
[18] J. Brehmer and W. Utschick, "Bottom-up Optimization of Layered Communication Systems based on Description-passing," Tech. rep., Inst. for Circuit Theory and Signal Processing, TUM-LNS-TR-05-02, Munich Univ. of Technology, June 2005.

BIOGRAPHIES

SHOAIB KHAN (khan@tum.de) received his B.S. degree from Bangladesh University of Engineering and Technology in 2001 and his M.S. degree from Technische Universität München (TUM) in 2003, both in electrical engineering. He is currently a member of research staff and Ph.D. candidate at the Media Technology Group of TUM. His research interests include cross-layer design, video compression, and wireless multimedia streaming.

MARCO SGROI (sgroi@docomolab-euro.com) is a researcher at NTT DoCoMo Communication Laboratories Europe. He received his Ph.D. and M.S. in electrical engineering and computer sciences at the University of California at Berkeley in 2002 and 1998, respectively, and his B.S. degree in electrical engineering at the Universita' di Roma La Sapienza in 1994. In 2003 he worked as a postdoctoral researcher at the University of California at Berkeley. His research interests include cross-layer optimization of network architectures and middleware for pervasive systems.

YANG PENG (yang.peng@tum.de) received his B.E. degree in information engineering from Tongji University, Shanghai, China, in 2001 and his M.S. degree in electrical engineering and information technology from TUM in 2004. He is currently a research staff member and Ph.D. candidate in the Department of Electrical Engineering and Information Technology of TUM. His research interests include cross-layer optimization for mobile multimedia, video compression, and streaming, as well as compression and transmission of image-based scene representations.

ECKEHARD STEINBACH [M'96] (Eckehard.Steinbach@tum.de) studied electrical engineering at the University of Karlsruhe, Germany, the University of Essex, Colchester, United Kingdom, and ESIEE, Paris, France. He received an engineering doctorate from the University of Erlangen-Nuremberg, Germany, in 1999. From 1994 to 2000 he was a member of research staff of the Image Communication Group at the University of Erlangen-Nuremberg. From February 2000 to December 2001 he was a postdoctoral fellow with the Information Systems Laboratory at Stanford University. In February 2002 he joined the Department of Electrical Engineering and Information Technology of TUM as a professor of media technology. His current research interests are in the area of networked multimedia systems.

WOLFGANG KELLERER [M] (kellerer@docomolab-euro.com) heads the Ubiquitous Services Platform group of NTT DoCoMo's European Research Laboratories, München, Germany. His research interests include mobile service platforms, peer-to-peer and sensor networks, and cross-layer design. He received his Dipl.-Ing. (M.Sc.) and Dr.-Ing. degree from TUM in 1995 and 2002, respectively. In 2001 he was a visiting researcher in the Information Systems Laboratory at Stanford University. He is a member of ACM.