Paradigms for Real-Time Video Communication and for Video Distribution

Ermin Kozica

KTH Electrical Engineering

Sound and Image Processing Laboratory
School of Electrical Engineering
KTH – Royal Institute of Technology

Stockholm 2011
Abstract

The use of new information technologies has drastically changed the way that we lead our lives. Communication technologies in particular have had a great impact on our day-to-day behavior. For example, it is now common to hear the voice and see the face of our loved-ones on another continents, or work with colleagues across the globe on a daily basis. With this change in behavior and the fast adoption of emerging technologies, new challenges in the telecommunications area are arising. This thesis is concerned with two such challenges: real-time video communication and video distribution.

The latency constraint in real-time video communication is in essence incompatible with the uncertainty of best-effort networks, such as the Internet. The recent arrival of smart-phones has added another requirement to the application, in terms of the limited computational and battery power. The research community has invested a large amount of effort in developing techniques that allow a mobile sender to outsource video encoding complexity to an unconstrained receiver by means of a feedback channel. We question that approach with respect to real-time applications, arguing that long round-trip-times may render any feedback unusable at best, and costly in practice. We investigate the effect of channel round-trip-times on the popular distributed video coding setup, as well as on the traditional hybrid video coding architecture. Using a simple analytical framework, we propose the use of systems that adapt to the video content and the network in real-time. Our results show that substantial improvements in video quality can be achieved when the feedback channel is used correctly.

The use of mobile devices has also a significant impact on the application of video distribution. In general, the multitude of devices that can be used to download and view video places new requirements on video distribution systems. The system must not only be able to scale to a large number of receivers in a bandwidth efficient manner, it must also support a wide range of network capacities and display capabilities. We address this problem by optimizing the set of rates that is used to provide video to receivers with heterogeneous requirements. Our approach is based on a favorable interpretation of the underlying mathematical problem, allowing the use of well-known quantization theoretic concepts. The resulting solution provides the possibility to design video distribution systems that adapt to changes in receiver characteristics online, with minimal delay.

Keywords: Video communication, packet-switched networks, real-time, online adaptation, heterogeneity.
List of Papers

The thesis is based on the following papers:


In addition to papers A-D, the following papers have also been produced in part by the author of the thesis:


Acknowledgements

The degree that I am about to receive is by no means the result of a one-man-show. First and foremost, I would like to acknowledge the contributions of my main advisor, Prof. Bastiaan Kleijn. Profound knowledge of the field, long experience and hard work ethics are truly great assets in any project. You provided those assets, and it is hard to imagine this thesis without your efforts on my behalf.

Approximately half-way through my studies, I was fortunate enough to get to know and do research with Prof. Kannan Ramchandran. Your innovative mindset, challenging attitude, and on-the-spot feedback were truly inspiring. Or was it just the California weather? In any case, this thesis is to a great extent the result of your contributions and I am thankful for those.

I would like to thank all the colleagues that I have had the pleasure of working with, both at KTH and at UC Berkeley. While space is limited to name you all, I would like to highlight the collaboration with Hao Zhang and Prof. Markus Flierl. I truly enjoyed our discussions and learned a lot from you two. Prof. Viktoria Fodor and Prof. Ragnar Thobaben, many thanks for proof-reading the thesis.

My research was tightly bound to a project financed by Global IP Solutions. I would like to extend my thanks to everyone involved in the project. In particular, Dr. Jan Lindén for enabling my visit to UC Berkeley, Dr. Jan Skoghund, Andrew MacDonald, Dr. Turaj Shabestary, Dr. Marco Paniconi, and Mikhal Shemer for making the US stay that much more interesting and productive, and Tina le Grand, Dr. Henrik Lundin, Dr. Björn Völcker, and Magnus Flodman for support in the Stockholm office.

The support that I have received outside of the office has been of great importance. I am thankful to my parents, Azra and Vehbo, for being there since I can remember, and to buraz Eldin and kum Edvin for all the experiences that we have had over the years.

Lejla, I met you at the start of my PhD studies. They are over now, but we will be together for ever. Thanks for taking my mind off of this silly research business every now and then.

Ermin Kozica, Stockholm, March 2011
Contents

Abstract i
List of Papers iii
Acknowledgements v
Contents vii
Acronyms xi

I Introduction 1
  1 Video Communication 3
  1.1 Source Coding 5
  1.2 Channel Coding 14
  2 Low-Complexity Video Encoding 17
  2.1 Complexity in State-of-the-Art Video Encoding 18
  2.2 Outsourcing of Encoding Complexity 20
  2.3 Complexity Outsourcing in Real-Time Communications 25
  3 Video Distribution to Heterogeneous Receivers 26
  3.1 One-to-Many Data Distribution in Theory 26
  3.2 One-to-Many Data Distribution in Practice 28
  3.3 Heterogeneous Receiver Requirements 30
  4 Summary of Contributions 32
References 34

II Included papers 47

A Frame-Bufferless Sum-Rate Constrained Video Encoding Using Feedback A1
  1 Introduction A1
**Bandwidth-Efficient Video Distribution Through Multi-Rate Video Encoding**

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction</td>
<td>D1</td>
</tr>
<tr>
<td>Optimization Framework</td>
<td>D4</td>
</tr>
<tr>
<td>2.1 Source Model</td>
<td>D4</td>
</tr>
<tr>
<td>2.2 Channel Model</td>
<td>D4</td>
</tr>
<tr>
<td>2.3 Receiver Assignment</td>
<td>D5</td>
</tr>
<tr>
<td>2.4 Optimization Objective</td>
<td>D6</td>
</tr>
<tr>
<td>Error-Resilience</td>
<td>D6</td>
</tr>
<tr>
<td>3.1 Automatic Repeat Query</td>
<td>D7</td>
</tr>
<tr>
<td>3.2 Forward Error Correction</td>
<td>D8</td>
</tr>
<tr>
<td>Analysis</td>
<td>D8</td>
</tr>
<tr>
<td>4.1 Visualization and Interpretation</td>
<td>D9</td>
</tr>
<tr>
<td>4.2 Automatic Repeat Query</td>
<td>D11</td>
</tr>
<tr>
<td>4.3 Forward Error Correction</td>
<td>D13</td>
</tr>
<tr>
<td>4.4 Solution Refinement</td>
<td>D16</td>
</tr>
<tr>
<td>Simulations</td>
<td>D17</td>
</tr>
<tr>
<td>Conclusion</td>
<td>D21</td>
</tr>
<tr>
<td>References</td>
<td>D21</td>
</tr>
</tbody>
</table>
# Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>3G</td>
<td>Third Generation Mobile Telecommunications</td>
</tr>
<tr>
<td>BEC</td>
<td>Binary Erasure Channel</td>
</tr>
<tr>
<td>BSC</td>
<td>Binary Symmetric Channel</td>
</tr>
<tr>
<td>CDN</td>
<td>Content Delivery Network</td>
</tr>
<tr>
<td>DCT</td>
<td>Discrete Cosine Transform</td>
</tr>
<tr>
<td>DPCM</td>
<td>Differential Pulse-Code Modulation</td>
</tr>
<tr>
<td>HD</td>
<td>High-Definition</td>
</tr>
<tr>
<td>HTTP</td>
<td>Hypertext Transfer Protocol</td>
</tr>
<tr>
<td>IP</td>
<td>Internet Protocol</td>
</tr>
<tr>
<td>IPTV</td>
<td>Internet Protocol Television</td>
</tr>
<tr>
<td>KLT</td>
<td>Karhunen-Loève Transform</td>
</tr>
<tr>
<td>LDPC</td>
<td>Low-Density Parity Check</td>
</tr>
<tr>
<td>MSE</td>
<td>Mean-Squared Error</td>
</tr>
<tr>
<td>P2P</td>
<td>Peer-to-Peer</td>
</tr>
<tr>
<td>TCP</td>
<td>Transmission Control Protocol</td>
</tr>
</tbody>
</table>
Part I

Introduction
Introduction

To an end user, a video is a video is a video. Whether it be stored on a computer, displayed in a real-time mobile video call, or streamed to millions of users. However, reality is that different applications have different physical conditions, limiting the achievable video quality, and requiring development of application-specific technologies. In this thesis, we address two research problems that deal with the design of video applications for challenging physical conditions. In the first research problem, we investigate how the lack of advances in certain hardware limits the quality of video applications. In particular, we investigate the effects of limited mobile-device computational and battery power on the quality of real-time one-to-one video communication. In the second research problem, we investigate how quality of video applications is limited by the advances in other hardware. In particular, we investigate how the heterogeneity of Internet peers, in terms of connectivity and communication devices, affects the performance of video distribution algorithms for large numbers of receivers.

The variety of devices that are available for communication over the Internet is illustrated in Figure 1. These devices differ with respect to their display and computational capabilities, ranging from mobile devices with constrained power-supply to television sets with HD resolution, as well as their communication link to the network, ranging from 3G and WiFi to broadband connections. Clearly, the variety in these two dimensions places demands on the network itself, and on the software that is used for communication.

Real-time one-to-one communication is expected to respect strict requirements on quality, in terms of audio-visual fidelity and conversation latency. The shift to mobile devices as the main communication medium has added further constraints to the equation. Scarce computational and battery power limit the currently achievable encoding performance. This is the case for video communication in particular, as the compression efficiency of standard video encoders is very much dependent on high computational complexity. In this thesis, the low-complexity video encoding problem is addressed by means of complexity-outsourcing through a feedback channel. While this approach is not new in video encoding, the thesis focuses on the
application of real-time video communication, where long channel round-trip-times potentially have a detrimental impact on performance. We show that realistic round-trip-times on the Internet channel make current solutions impracticable. Methods that adapt to the channel round-trip-time and use the feedback channel only to the extent that it is advantageous from a compression efficiency point of view are proposed.

An effect of the fast adoption of information technology is that now everyone can, and to a large extent does, produce, publish and distribute multimedia content at extremely low costs. The sheer amount of available content is challenging the capabilities of our main distribution medium, the Internet. Efficient distribution algorithms are necessary to ensure the best use of its physical capacity, especially for video content, as the data-rates here are significantly larger than for, e.g., speech and audio content. At the same time, the numerous end-users that have access to the content have heterogeneous needs as they cover a wide spectrum of communication devices and channels. In this thesis, we analyze two popular content distribution technologies, which use multi-rate encoding to ensure adaptivity to heterogeneous receiver requests. These are Content Delivery Networks (CDNs) and IP Multicast. CDNs have recently made an entrance to video distribution applications by means of HTTP based streaming solutions, while IP Multicast has found its main application in private networks, such as IPTV.
networks, where protocol support in network routers can be enforced. We propose an innovative and useful interpretation of the multi-rate video encoding optimization problem in these technologies. This interpretation leads to the (rather) straight-forward application of tools developed in quantization theory.

The thesis introduction is organized as follows. Section 1 introduces the reader to digital video communication systems and the criteria used in their design. Section 1 also presents a selection of useful information theoretic results, which are needed for the understanding of the remainder of the thesis. Methods for data compression are emphasized, as it turns out that all problems that we solve can be thought of as specific instantiations of the data compression problem. Sections 2 and 3 introduce the reader to the two specific problems of low-complexity video encoding and bandwidth-efficient video distribution. The problems are presented as constrained versions of the digital video communication system introduced in Section 1.

1 Video Communication

The components of a general digital communication systems are shown in Figure 2. A source produces the data that is to be communicated to the sink over the given channel. The goal of the designer of the communication system is that of ensuring that the communication is good enough as measured by some data fidelity measure, which compares the original data and the data that is reconstructed at the sink, and a measure of cost, such as the amount of data that needs to be communicated.

![Source, channel, sink and the digital communication system.](image)

The goal of the digital communication system designer is to enable the reconstruction of the source data at the sink, optimized with respect to the amount of information transmitted over the channel and a data fidelity measure.
The design of communication systems is made difficult by two points of uncertainty. These are the source and the channel. Common for all sources is that there is uncertainty with the respect to the data that it produces, such that communication is actually necessary. Clearly, depending on the predictability of the source and the desired data fidelity, different amounts of data need to be transmitted to achieve this goal. The channel, on the other hand, introduces uncertainty in that its output is not necessarily equal to the input. Thus, measures against data corruption have to be taken.

Shannon suggested in 1948 a modularized solution, which is optimal in the limit of infinite delay, to the problem of efficient communication [110]. Shannon’s modularized system is illustrated in Figure 2. The uncertainty of the source is handled by the source encoder, whose goal is to represent the source data as efficiently as possible. The uncertainty of the channel is handled by the channel encoder, whose goal is to protect the compressed data, such that errors introduced by the channel can be corrected.

As mentioned above, Shannon derived the source-channel separation principle with no consideration of the communication delay. In most practical systems, where the acceptable communication delay is constrained, these results are not applicable. Examples where joint source-channel coding for video is addressed include [48, 66, 101, 105, 144]. Nevertheless, the proposed modularization has the practical advantage that it allows for the independent development of general purpose source and channel encoding and decoding algorithms, coders in short, which can be applied to a large number of applications. Further, as this thesis shows, these independently developed algorithms can be applied to topics of interest in video communication that are not in the standard communication problem. Therefore, the first part of the introduction focuses on introducing underlying theory and practical methods for the independent design of source and channel coders.

The source is commonly modeled as a random process $X$ that is either continuous in time, e.g., speech, audio and video, or discrete in time, e.g., English text. One important result in signal processing is the sampling theorem. Nyquist and Shannon showed in [88] and [111], respectively, that perfect reconstruction of a continuous-in-time source $X$ from a discrete set of samples is possible if the source outputs a bandlimited signal and the sampling frequency is double the highest signal frequency. Based on this theorem, continuous sources are sampled and thus discretized in most modern communication systems. In the remainder of this introduction, we present methods for communication of discrete sources only.

Let the realizations of the discrete source $X$ take values from an alphabet $\mathcal{A}$, which may have finite or infinite cardinality. Let us denote by $x^m = (x_{t-m+1}, x_{t-m+2}, \cdots, x_t)$ the realizations of $X$ at time instances $t-m+1, t-m+2, \cdots, t$. We write $x^m \in \mathcal{A}^m$. A source encoder $i = f^\mathcal{S}(x^m)$ is a function of the source output $x^m$, mapped to a finite set $\mathcal{I}$. Source coders are scalar if they operate in one dimension with $m = 1$, and vector
The elements of the set $\mathcal{I}$ are represented by codewords that consist of zeros and ones. We use the notation $\ell(i)$ to denote the length of codeword $i \in \mathcal{I}$. Note that the codeword lengths need not be equal. The source decoder $g^S$ is a mapping from $\mathcal{I}$ to $\mathcal{A}^m$, which is possibly different from $\mathcal{A}^m$. We denote the source decoder output for input $i = f^S(x^m)$ by $\hat{x}^m = g^S(i)$. The notation $q = g^S \circ f^S$ is often used, such that $\hat{x}^m = q(x^m)$.

Before the source decoder can be put into use, the signal that is the binary representation of codewords in $\mathcal{I}$ has to be transmitted over the channel. Let $u^k$ be the $k$-bit concatenation of the binary representation of a number of codewords from $\mathcal{I}$. The channel encoder $f^C$ is designed to ensure error-free transmission of $u^k$ over the channel in the limit of $k \to \infty$. The encoder introduces redundancy in the signal to be transmitted by mapping $u^k \in \{0,1\}^k$ to a codeword $v^n \in \{0,1\}^n$, where $n \geq k$. There are $k$ codewords in total. A specific class of channel codes are called **systematic channel codes**. These produce codewords $v^n$ that are the concatenation of the input $u^k$ and a number of **parity** bits. Due to channel errors, the receiver does not receive signal $v^n$, but an erroneous version $\tilde{v}^n$. It is the task of the channel decoder $g^C$ to correct the errors in the received signal $\tilde{v}^n$ and reproduce the channel encoder input $u^k$. The independent design of source and channel coder design relies on the assumption that the channel coder can reproduce $u^k$ with high probability for finite $k$.

The remainder of the section introduces the source and channel coding problems, the basic theory governing achievable performance of source and channel coders and some practical design aspects. The focus lies on those concepts that are useful for understanding the contributions of this thesis. In particular, the methods described for source coding are directly applicable in the design of low-complexity video encoders in Paper A and Paper B. Further, those same source coding methods are used in the seemingly unrelated context of optimization of multi-rate video encoders in bandwidth efficient video distribution, which is addressed in Paper C and Paper D. Methods for channel coding are not central in this thesis, but an overview is provided as the low-complexity video encoder in Paper A is based on the use of channel codes for data compression.

### 1.1 Source Coding

The objective of source coding is the efficient representation of the source output at a fidelity that is sufficient for the application at hand. With respect to the introduced notation, the description efficiency is quantified in the number of bits, rate $R$, that is necessary to describe source realizations. The rate is measured as an expectation of the codeword length $\ell(i)$

\[
R = \sum_{i \in \mathcal{I}} p_f(i) \ell(i), \tag{1}
\]
over the probability distribution \( p_I(i) \), where \( I \) is a underlying random variable that is a function of the source encoder and the statistical properties of the source. The codeword lengths of a uniquely decodable code have to satisfy Kraft’s inequality \([67]\), which leads to the following bound on the achievable rate

\[
R \geq H(p_I) = -\sum_{i \in \mathcal{I}} p_I(i) \log_2 p_I(i). \tag{2}
\]

The expression for the bound \( H(\cdot) \) is called the entropy function, and is central in information theory\(^1\).

The source coding objective of efficient and precise source representation can be achieved by the exploitation of redundancy and/or irrelevancy. Redundancy is the dependence between source samples, and it can be exploited without loss of information, so called lossless source coding \([43, 53, 104, 115, 146]\). Irrelevancy are those variations in source data that are irrelevant to the receiver, and its exploitation is done with loss of information. Source coding methods that do introduce errors in the reconstruction \( \hat{x}^m \) with respect to \( x^m \) are called lossy \([10, 45, 59, 90, 112, 128]\). There exists a trade-off between the reconstruction error, distortion, and the encoding rate. The most commonly used measure of distortion is the mean squared-error (MSE) distortion

\[
d(x^m, \hat{x}^m) = \frac{1}{m} \|x^m - \hat{x}^m\|^2. \tag{3}
\]

In summary, the source coding objective can compactly be written as

\[
\begin{align*}
\min_{q, \ell} & \quad \mathbb{E}_{X^m} [d(X^m, q(X^m))] \\
\text{s.t.} & \quad \mathbb{E}_I [\ell(I)] \leq \bar{R},
\end{align*}
\]  \( \tag{4} \)

where \( \bar{R} \) is the constraint put on the encoding rate and \( X^m \) represents the random variable that is a vector of \( m \) samples from \( X \). Note that the distortion measure \( d \) is not limited to the MSE. In fact, the analysis of video distribution technologies in Paper C and Paper D is based on a different measure of distortion.

**Rate-Distortion Theory**

Analysis of sources and the design of compression algorithms for these require the assumption of a source model. The most general model is that of a random process \( X \), with no constraints on its composition. Shannon showed in \([112]\) that the source coding of such processes can be characterized by a rate-distortion function \( R(D) \), which gives a tight lower bound on the rate \( R \) that is necessary to describe the source with at most the
distortion $D$. Shannon’s results are derived in the limit of infinite coding delay, with $m \to \infty$, and for difference distortion measures, where $d(x^m, \hat{x}^m) = d(|x^m - \hat{x}^m|)$. The requirement of infinite coding delay is the largest limitation of Shannon’s work when it comes to practical systems. Nevertheless, the rate-distortion is very useful as the benchmark for any source coder implementation.

For specific sources, the rate-distortion curve can be derived by first finding a lower bound on it, referred to as the Shannon lower bound [75,112], and then showing that the bound coincides with the rate-distortion function. While this procedure is not always possible, some properties about the rate-distortion curve are known to be true in general. The rate-distortion curve has the intuitive properties that it is non-increasing, thus giving diminishing returns, and convex. To see this, consider two coders that produce two points on the rate-distortion curve $R_1(D_1)$ and $R_2(D_2)$, with $D_2 > D_1$. The rate-distortion function must be non-increasing with $R_1(D_1) \ge R_2(D_2)$. The contrary would indicate that $R_2(D_2)$ is not on the rate-distortion curve. Further, the rate-distortion curve is convex, as any point on the line connecting points $(R_1, D_1)$ and $(R_2, D_2)$ can be reached by appropriately weighted time-sharing between the two coders, thus forming a new coder.

Figure 3 shows the rate-distortion curve for the independent identically distributed (iid) Gaussian source with variance $\sigma^2 = 1$. Gaussian iid sources output independent scalar realizations drawn from the Gaussian probability

\[1\text{The entropy function is closely related to entropy as it is used in statistical physics} \]
distribution
\[ f_X(x) = \frac{1}{\sqrt{2\pi}\sigma^2} \exp\left\{-\frac{1}{2\sigma^2} (x - \mu)^2\right\}, \]  
with mean \( \mu \) and variance \( \sigma^2 \). The rate-distortion function is given by
\[ R(D) = \begin{cases} \frac{1}{2} \log_2 \frac{\sigma^2}{D} & \text{if } D < \sigma^2 \\ 0 & \text{otherwise}. \end{cases} \]  

While the rate-distortion curve can be analytically found only for certain sources, Blahut derived in [13] an algorithm, known as the Blahut algorithm, that can be used to compute the rate-distortion curve numerically. If the reproducing alphabet \( \hat{A} \) is finite, the Blahut algorithm converges to a point on the rate-distortion curve [20]. The reader is referred to [13] for details on the Blahut algorithm.

**High-Rate Quantization**

In general, the source coding optimization problem posed in (4) is non-convex and difficult to analyze. One approach to solving the source coding problem, is that of assuming a high encoding rate in expression (1), such that a favorable approximation to a convex problem with an analytical solution can be made. Analytical solutions are also attractive for applications where the source statistics are dynamic and fast adoption of the quantizer is desired.

Let us define the quantization cell as a partition of the space \( \mathcal{A}^m \)
\[ \text{cell}(\hat{x}^m) = \{x^m \in \mathcal{A}^m : q(x^m) = \hat{x}^m \in \hat{A}^m\}. \]  

The high-rate assumption is the assumption that the probability density function of the source does not change appreciably within the quantization cell, \( f_{X^m}(x^m) \approx f_{X^m}(\hat{x}^m) \) for all \( x^m \in \text{cell}(\hat{x}^m) \). It was first used by Bennett in [9] to find an approximation to the mean-squared error of quantized signals in one dimension. Approximations based on the high-rate assumption are valid in the limit of cell volumes going to zero.

Using the high-rate assumption, the objective function in (4) can for the \( \text{MSE} \) distortion measure be reformulated as
\[ E_{X^m} \left[ d(X^m, q(X^m)) \right] \]
\[ \approx \sum_{\hat{x}^m \in \hat{A}^m} f_{X^m}(\hat{x}^m) \int_{\text{cell}(\hat{x}^m)} \frac{1}{m} \|x^m - \hat{x}^m\|^2 dx^m \]
\[ = \sum_{\hat{x}^m \in \hat{A}^m} p_I(f^S(\hat{x}^m)) \mathbb{V}(\hat{x}^m) \hat{H} C(\hat{x}^m), \]
where we defined the cell volume

\[ V(\hat{x}^m) = \int_{\text{cell}(\hat{x}^m)} dx^m, \]  

(11)

and the coefficient of quantization

\[ C(\hat{x}^m) = V(\hat{x}^m) - \frac{1}{m} \int_{\text{cell}(\hat{x}^m)} \frac{1}{m} ||x^m - \hat{x}^m||^2 dx^m. \]  

(12)

The effect of the cell shape on the expected distortion is measured by the coefficient of quantization and it is independent of rate.

In one dimension, \( m = 1 \), there is only one possible cell shape, giving \( C(\hat{x}) = \frac{1}{12} \) for all cells and the approximate distortion

\[ \frac{1}{12} \sum_{\hat{x}^m \in A^m} p_I(fS(\hat{x}^m))V(\hat{x}^m)^2 \approx \frac{1}{12} \int_A f_X(x)V(x)^2 dx. \]  

(13)

Zador showed in [136] that the optimal coefficient of quantization equals \( \frac{1}{12\pi} \) in the limit \( m \to \infty \). This result is useful in the evaluation of the performance of scalar and vector quantizers.

As the optimal cell shape is not known in most dimensions, Gersho used the conjecture [36] that the cells in higher dimensions form a tessellation of the space \( A^m \) for which the coefficient of quantization is constant \( C(\hat{x}^m) = \bar{C} \). Gersho’s conjecture gives the approximate distortion expression

\[ C \sum_{\hat{x}^m \in A^m} p_I(fS(\hat{x}^m))V(\hat{x}^m)^2 \approx \bar{C} \int_{A^m} f_X(x)V(x)^2 dx. \]  

(14)

Commonly, equations (13) and (14) are rewritten as a function of a distribution of cells in \( A^m \), also known as the reconstruction point density, using the definition

\[ g(x^m) \approx \frac{1}{V(\hat{x}^m)}, \]  

(15)

for any \( x^m \in \text{cell}(\hat{x}^m) \). Then, an analysis is performed on the asymptotic behavior of the optimization objective function by design of the reconstruction point density \( g(\hat{x}^m) \). This approach was first used by Panter and Dite in [93] for the scalar case \( m = 1 \).

The solution to (4) can be obtained analytically if the constraint also is expressed as a function of the reconstruction point density \( g(\hat{x}^m) \). Instead of optimizing the mapping \( \ell \) in the quantizer design, the constraint in (4) is often exchanged for either one of two constraints. One constraint is that all codewords should have the same codeword length by design

\[ \ell(i) = \lfloor \log_2(|\mathcal{I}|) \rfloor, \quad \text{for all } i \in \mathcal{I}, \]  

(16)
giving resolution-constrained quantization. The other constraint is that the codewords should have a bounded entropy

$$-E[I \log_2 p_I(I)] \leq \bar{R},$$

(17)

thus giving entropy-constrained quantization. A lossless variable length code is applied to the quantization indices of entropy-constrained quantizers, e.g. a Huffman code [53]. As the expected codeword length according to Kraft’s inequality is greater than or equal to the entropy, it is obvious that resolution-constrained quantizers are less general and thus yield codes with longer average codeword length.

Resolution-constrained quantizers are useful in some systems where fixed codeword length is required. For example, in systems with fixed bit-rate. For the purpose of this thesis resolution-constrained quantizers are of specific interest as the optimization of feedback rate in Paper A, as well as the optimization of bandwidth efficient video distribution in Paper C and Paper D, can be formulated in the form of a quantizer optimization problem with a constraint on the codeword resolution. For that reason, we give here a short derivation of the reconstruction point density that minimizes the distortion under the resolution-constraint, followed by an explanation of how the actual reconstruction points can be obtained from this density.

The total number of cells in a resolution-constrained quantizer is obtained by integrating the reconstruction point density over the space $A^m$.

Thus, the optimization constraint becomes

$$\int_{A^m} g(x^m) dx^m \leq |\mathcal{I}|,$$

(18)

where $|\mathcal{I}|$ is the cardinality of the set $\mathcal{I}$. Using the method of Lagrange multipliers, it is possible to derive the reconstruction point density

$$g(x^m) = |\mathcal{I}| \frac{f_{X^m}(x^m)}{\int_{A^m} f_{X^m}(x^m) dx^m}$$

(19)

which is asymptotically optimal with respect to the optimization objective (14)—using (15)—and under the resolution constraint (18).

To obtain the reconstruction points from $g(x^m)$, it is possible to use a method inspired by Bennett’s one-dimensional companding quantizer [9]. Bennett suggested the implementation of a non-uniform quantizer by the transform of the signal $x$ with a monotonic compressor function, followed by a uniform quantizer. The advantage lies in the low complexity of the implementation of uniform quantization, especially in higher dimensions. The inverse of the compressor function, the expander, is used to map reconstruction points in the compressed domain to the original signal domain.
For dimensions larger than one, Gersho showed in [36] that the companding quantizer incurs a loss compared to optimal quantization.

For this thesis, the location of the reconstruction points in the original domain is of interest, rather than the implementation of a quantizer. This is a problem that is similar to that of histogram matching, a parallel that was recognized by Hummel in [54]. Thus, Paper C and Paper D all derive the expander that maps uniformly distributed points to reconstruction points distributed according to $g(x^m)$.

### Necessary Conditions for Quantizer Optimality

The high-rate analysis allows for analytical design of quantizers and prediction of their performance. However, the high-rate assumption is not of sufficient accuracy for all applications and other approaches to quantizer design are needed. For example, state-of-the-art video coders use well-tuned quantization maps as a means to distribute the rate among different variables that are to be quantized, although it is not necessary at high rates (as will be described in a separate section on rate allocation).

One approach to handle low rates is that of deriving the necessary conditions for quantizer optimality from (4), and designing algorithms that give quantizers that fulfill these conditions [74,79,83]. As in the high-rate analysis, the quantizer is typically chosen to be resolution-constrained or entropy-constrained by design. In this introduction, we focus on resolution-constrained quantization, where the codeword length $\ell(i)$ is equal for all $i \in I$. To ensure efficient representation of the codewords, $|I|$ is typically chosen to be a power of 2, such that $\ell(i) = \log_2 |I|$.

The optimization objective in (4) is that of finding the optimal quantization mapping $q$, consisting of the appropriate codomain $\hat{A}^m$ and the quantization cells $\text{cell}(\cdot)$. The expected distortion can equivalently be expressed as the integral over the space $A^m$ or the sum of distortion contributions of each cell

$$E_{X^m}\left[d(X^m, q(X^m))\right] = \int_{A^m} f_{X^m}(x^m) d(x^m, q(x^m)) dx^m$$

$$= \sum_{\hat{x}^m \in \hat{A}^m} \int_{\text{cell}(\hat{x}^m)} f_{X^m}(x^m) d(x^m, \hat{x}^m) dx^m. \quad (22)$$

The necessary conditions for optimality of $\text{cell}(\cdot)$ is that $x^m$ is mapped to the reconstruction point in $\hat{A}^m$ that minimizes the expression $d(x^m, q(x^m))$. Thus the necessary condition for cell optimality is

$$\text{cell}(\hat{x}^m) = \{x^m \in A^m : d(x^m, \hat{x}^m) \leq d(x^m, y^m), y^m \in \hat{A}^m\}.$$ 

(23)
The necessary condition for the optimality of the reconstruction points in $\hat{A}_m$ is

$$\hat{A}_m = \{ \hat{x}_m^m \in \mathcal{A}_m : \hat{x}_m^m = \arg\min_{\hat{x}_m^m \in \mathcal{A}_m} \int_{\text{cell}(\hat{x}_m^m)} f_X(x_m^m) d(x_m^m, \hat{x}_m^m) dx_m^m, \} ,$$

(24)
such that the distortion contribution of each cell is minimized in (22).

The Lloyd-Max algorithm, independently introduced in [79] and [83], is based on iterative updates according to (23) and (24). In each iteration, (i) the cell partitioning is optimized with (23) while keeping the reconstruction points $\hat{A}_m$ fixed, and (ii) the reconstruction points are optimized with (24) while keeping the cell partitioning fixed. The algorithm converges to a local minimum as each step decreases the value of the optimization objective function (20). Clearly, the initialization point of the algorithm will affect its final performance. One commonly used algorithm was proposed in [74], where codebook training is performed for codebooks of increasing size and the initialization of each is based on the termination point for a smaller codebook.

**Low-Complexity Exploitation of Source Memory**

The source coding results described so far have been general in that they have been formulated for any dimensionality of the input $x_m$. According to high-rate analysis vector quantizers have three advantages over scalar quantizers: the space filling advantage, the memory advantage, and the shape advantage [80]. To get a sense for their importance it is considered worth to mention that the space filling advantage is always present and upper-bounded by 0.254 bits per source sample (due to a decreasing coefficient of quantization), the memory advantage is non-existent if the source is iid, and the shape advantage exists only for constrained-resolution quantization, depends on the source distribution and is non-existent for uniform sources.

Although vector quantizers possess these advantages, they have some practical drawbacks. For example the complexity of resolution-constrained vector quantizers, in terms of the requirement of computational resources of the coder, grows exponentially with dimensionality $m$, unless fast decoding can be used as in the case with structured quantizers [19] or decoding complexity is constrained [14]. In both cases, loss in performance is implied due to considered complexity costs. There exist other methods for exploitation the vector advantages, mostly focused on the memory advantage. Two of the most common methods are transform coding and predictive coding. Both these methods are used in the video coding algorithms that will be described in more detail in Section 2.

In transform coding, the source samples $x_m$ are preprocessed by a function, followed by low-complexity scalar quantization of the transform coefficients. Commonly the transform is designed to give critical sampling,
such that the coefficients vector, say \( c^m \), is of the same length as \( x^m \). The transform function is commonly taken to be linear, such that

\[
c^m = T x^m,
\]

where \( T \) is a \( t \times t \) transform matrix. For the described source coding setup, with scalar quantization of \( c^m \), Huang and Schultheiss showed in [52] that the Karhunen-Loève transform (KLT), which produces \( c^m \) with a diagonal covariance matrix, is optimal for multivariate Gaussian sources. The KLT makes the transform coefficients \( c^m \) independent, and the transform coding setup, combined with rate distribution based on waterfilling [62], nullifies the memory advantage of vector quantization. Rate distribution among several quantizers, such as those of a set of transform coefficients, will be further explained in the following section.

While the KLT is the optimal transform, it has the two disadvantages that it is data-dependent and that its calculation requires \( m^2 \) operations. In many practical applications, calculating the KLT is too costly and other, data-independent, transforms that require \( t \log t \) operations are preferable [57, 95]. Ahmed, Natarajan and Rao introduced in [4] the discrete cosine transform (DCT), as an approximation to the KLT. The DCT approximates the KLT well for first-order Markov signals with high correlation and converges to the KLT asymptotically in \( m \) [57]. For further details on transform coding, the reader is directed to the overview paper by Goyal [44].

In predictive coding, the source memory is exploited by prediction of the considered sample and encoding only of the prediction error. Closed-loop prediction is commonly used, where the prediction is based on quantized source outputs \( \hat{x}_t \). Using the predictor function \( \varphi(\cdot) \) closed-loop prediction is written

\[
\tilde{x}_t = \varphi(\hat{x}_{t-1}, \hat{x}_{t-2}, \cdots).
\]

Predictive coding was introduced by Cutler in [21], where he patented closed-loop differential pulse code modulation (DPCM) with \( \varphi(\cdot) \) being a linear function of prediction errors \( \{x_t - \hat{x}_{t-1}, x_{t-1} - \hat{x}_{t-2}, \cdots\} \). Elias studied the optimization criterion for the predictor function in source coding, and showed that it is the entropy of the prediction error that is to be minimized [26, 27]. Recently, it was shown that predictive coding can reach the rate-distortion function of Gaussian variables [137]. However, causal prediction, as in (26), costs due to the space filling advantage of vector quantization [76, 137] 0.254 bits per source sample in compression efficiency with respect the rate-distortion curve.
Rate-Allocation for a Set of Quantizers

In the application of above presented theory to real sources, the problem of joint optimization of several quantizers often appears. For example, when coding a source by using a transform followed by scalar quantization of the transform coefficients, it is necessary to optimize the quantizers of all transform coefficients jointly.

Let \( X^m = \{X_{t-m+1}, X_{t-m+2}, \ldots, X_t\} \) be a vector of \( m \) random variables, which are to be encoded with the source coding objective in (4). Consider the case where the encoding of \( X^m \) is to be done independently for each element in the vector. If the distortion measure can be rewritten as a sum of distortions for each element, such as for the MSE distortion measure (3), the quantization optimization objective (4) becomes

\[
\min_{q, \ell} \sum_{j=t-m+1}^{t} E_{X_j} [d(X_j, q_j(X_j))] \\
\text{s.t.} \quad \sum_{j=t-m+1}^{t} E_{I_j} [\ell(I_j)] \leq \bar{R},
\]

(27)

where \( q_j \) denotes the \( j \)th out of \( m \) scalar quantizers. The unconstrained problem, using Lagrange multiplier \( \lambda \), is given by

\[
\min_{q, \ell} \sum_{j=t-m+1}^{t} E_{X_j} [d(X_j, q_j(X_j))] + \lambda \sum_{j=t-m+1}^{t} E_{I_j} [\ell(I_j)] \\
= \min_{q, \ell} \sum_{j=t-m+1}^{t} \left( E_{X_j} [d(X_j, q_j(X_j))] + \lambda E_{I_j} [\ell(I_j)] \right).
\]

(28)

The term \( E_{X_j} [d(X_j, q_j(X_j))] + \lambda E_{I_j} [\ell(I_j)] \) is usually referred to as the Lagrangian cost function of the quantizer \( q_j \). The solution to the unconstrained problem, for a given \( \lambda \), is the same as the solution to the constrained problem with the rate constraint that equals the sum of all rates in the unconstrained solution [114]. Thus, joint optimization of the quantizers can be performed by optimization of the unconstrained problem, which in turn can be solved independently for each quantizer using the Lagrangian cost function, which is the weighting of the expected distortion and the expected rate for that quantizer.

The above separation of the joint optimization problem is often used in video coding, as will be explained later. Two examples are the rate-allocation between transform coefficient quantizers [114], and rate-allocation between independently coded macro-block in a video frame [39,118,119].

1.2 Channel Coding

The objective in channel coding is the efficient and error-free transmission of the data \( u^k \) from the sender to the receiver, by representation of \( u^k \) by \( v^n \), where \( v^n \in C \subseteq \{0, 1\}^n \). With respect to the introduced notation, the efficiency of transmission, under the constraint of zero transmission errors
in the limit $k \to \infty$, is measured in terms of the channel coding rate, $\rho = \frac{k}{n}$, which represents the fraction of transmitted information bits and total bits. On an error-free channel, the rate 1 is easily obtained. The question in channel coding design is how to achieve positive, and preferably high, rates for channels that introduce errors, erasures or distortion in the transmission.

The Channel Coding Theorem

To perform analysis of channels and design efficient channel codes, a source model has to be assumed. The channel is commonly characterized by a conditional probability function $p_{\tilde{V}^n|V^n}(\tilde{v}^n|v^n)$, where $V^n$ and $\tilde{V}^n$ are the underlying random variables of $v^n$ and $\tilde{v}^n$. Shannon’s channel coding theorem [110] states that the channel capacity

$$C = \max_{p_{V^n}(V^n)} H(p_{V^n}) - H(p_{V^n|\tilde{V}^n})$$

represents the maximum achievable channel code rate. While Shannon did not provide a rigorous proof to the channel coding theorem, many have been given since 1948. A commonly quoted paper is that of Gallager in [34].

A closed-form expression for the channel capacity is not known for general channels. A special class of channels is the class of discrete stationary memoryless channels with iid realizations from a fixed distribution

$$p_{\tilde{V}^n|V^n}(\tilde{v}^n|v^n) = \prod_{j=1}^{n} p_{\tilde{V}|V}(\tilde{v}_j|v_j).$$

This class of channels is easier to analyze, and can be used to model the channels in many practical applications. Two specific memoryless channels are the binary symmetric channel (BSC) and the binary erasure channel (BEC), whose conditional probability functions are given in Figure 4, where $\alpha$ is a channel parameter. Shannon used symmetric channels as an example in [110] and derived their capacity. For the special case of the BSC, the capacity is $1 - H(p_{\tilde{V}|V})$. The capacity of the BEC is $1 - p_{\tilde{V}|V}(\varepsilon|0)$, as given in, e.g., [28].

Arimoto and Blahut derived an algorithm that can be used to find the channel capacity numerically in those cases analysis is not possible. The reader is referred to [6,13] for details.

The channel coding theorem is proved and valid in the limit of infinite coding delay, with $n \to \infty$. This is comparable to the results of rate-distortion theory in source coding. In fact, the two problems are functional duals, as recognized by Shannon [110]. The infinite coding delay is again a limitation when it comes to practical systems, but the channel coding theorem is very useful as the benchmark for any specific channel coder implementation.
Linear Codes and Bounded Distance Decoding

In practical code design, encoding and decoding complexity is an issue of much importance. A simple measure of complexity is the amount of memory needed to define and store a code. As the number of codewords grows exponentially with \( k \), just storing a codebook of all possible codewords \( u^k \) quickly becomes impractical.

A special class of codes are linear codes, which have codewords that are in a linear subspace \( C \) of \( \{0,1\}^n \). Thus, any linear combination of two codewords gives another codeword in \( C \). Linear codes do not require the storage of the codebook, as encoding can be performed online with a task that requires \( O(nk) \) operations. The first linear code was the Hamming code, introduced in [46]. It has been shown that linear codes can achieve the capacity for binary memoryless symmetric channels such as the BSC and the BEC, e.g., [22].

Let us define the Hamming distance \( d_H(v^n_i, v^n_j) \) of two binary codewords \( v^n_i \in C \) and \( v^n_j \in C \) as the number of positions in which the codewords differ. The expression \( \min d_H(v^n_i, v^n_j) \) gives the minimum distance of the code, where the minimization is performed over all distinct codeword pairs. A linear code can correctly decode a codeword with up to \( \lfloor \min d_H(v^n_i, v^n_j) - 1 \rfloor \) errors or \( \min d_H(v^n_i, v^n_j) \) erasures. Reed and Solomon introduced in [102] the optimal linear code—commonly referred to the Reed-Solomon code—with respect to the minimum distance of the code.

A maximum likelihood decoder decodes \( \tilde{v}^n \) to the codeword \( v^n \) that maximizes \( p_{V|\tilde{V}}(v^n|\tilde{v}^n) \). For the binary symmetric channel, this is equivalent to the codeword \( v^n \) that minimizes the Hamming distance \( d_H(\tilde{v}^n, v^n) \). Unfortunately, maximum likelihood decoding is an NP complete problem for linear codes.

In search for reasonable decoding complexity, the research community
focused for a long time on the alternative decoding criteria of bounded distance decoding, which allows for low-complexity, but suboptimal, decoding for certain linear codes [11,82]. A bounded distance decoder decodes \( \tilde{v}^n \) to the valid codeword \( v^n \) that minimizes \( d_H(\tilde{v}^n, v^n) \) only if there are no more than \( \tau \) errors, where \( \tau \leq \left\lfloor \frac{\min_{i,j} d_H(v^n_i, v^n_j) - 1}{2} \right\rfloor \) [117].

Iterative Decoding

Bounded distance decoding offers a solution to the problem of sufficiently low decoding complexity. However, the performance of codes that are optimal with respect to bounded distance decoding do not achieve rates at the capacity. A proof can be found in [103], and is easily obtained by putting an upper bound on the performance of bounded-distance-optimal codes and showing that the bound is strictly smaller than channel capacity. Thus, bounded distance decoding is not a good substitute criterion for maximal likelihood decoding.

Gallager introduced in his PhD thesis [33] iterative decoding of a special class of linear codes, named low-density parity check (LDPC) codes, for which approximate maximum likelihood decoding is possible at, by today’s standards, a reasonable decoding complexity. LDPC codes approach the channel capacity set forth by Shannon in 1948 [110] and have received much attention in the past decade. Another class of codes that approach the channel capacity is Turbo codes, which were introduced by Berrou, Glavieux and Thitimajshima in 1993 [12]. Both LDPC and Turbo codes use iterative belief propagation algorithms for decoding and error correction.

2 Low-Complexity Video Encoding

The existence of low-complexity video encoding algorithms is becoming increasingly important as the connection speed of mobile networks approaches the bit-rates that are sufficient for reasonable video quality. However, the video quality at these bit-rates is in state-of-the-art coders achieved by means of a high-complexity video encoder paradigm. Thus, direct implementation of state-of-the-art encoders is (i) not a viable alternative for real-time applications, and (ii) prone to fast depletion of battery energy on mobile devices. In this section, we give a description of these state-of-the-art video coding algorithms, and their use of general source coding methods that were introduced in Section 1. We argue that these algorithms are not suitable for implementation on mobile devices. Then, we present the idea of feedback-enabled complexity outsourcing as it has been described in the literature and is further studied in Paper A and Paper B.
2.1 Complexity in State-of-the-Art Video Encoding

The video coding problem is a specific incarnation of the source coding problem and video coders are based on the theory developed for the coding of general sources. Digital video data is sampled in time, resulting in a series of frames (or images), and in space. In the context of previous definitions, the block of source realizations $x^n$ denotes $n$ frames, where the frame $x_t$ sampled at time $t$ is a vector variable holding the values of the frame on a spatial sampling grid. The video encoder compresses the data by exploiting redundancy and irrelevancy in the temporal and spatial dimensions.

Consider the encoding of a frame $x_t$. To exploit local data dependencies and reduce system memory requirements, it is common [15, 55, 56, 86] to divide $x_t$ into macro-blocks that are encoded in raster-scan order. Let $x_t(r, c)$ denote the $B \times B$ macro-block, whose top-left pixel is located on the $r$th row and the $c$th column on the spatial sampling grid of frame $x_t$. The macro-block data is correlated to spatially and temporally close macro-blocks.

The hybrid video coding setup is based on DPCM-like temporal and/or spatial prediction of the macro-block that is to be encoded, followed by encoding of the prediction residual with transform-based exploitation of spatial redundancy. The prediction gain of temporal macro-block prediction was studied in for example [31, 38, 40], while spatial prediction was introduced only with H.264 [56, 140]. However, the dependencies in video are not stationary and there exists no good model that allows for open-loop exploitation of them. Therefore, state-of-the-art video coders allow for the use of many different macro-block dependency models, so called modes, as a means to increase robustness of compression against modeling mismatches [91, 119]. Assuming that the macro-blocks can be encoded independently without loss of compression efficiency, it is possible to use the rate-allocation methodology to ensure optimal rate-allocation for all macro-blocks [39, 114, 118] as indicated in (28). That is, the encoding of each macro-block can be performed using the Lagrangian cost function $J(x_t(r, c)) = D(x_t(r, c), \hat{x}_t(r, c)) + \lambda R(x_t(r, c))$. However, even in this case the encoding complexity is too high for real-time applications.

A large number of research papers address the high complexity of hybrid coders. Examples that target the computational efficiency of H.264 encoders in particular are given by [70, 92, 126, 134]. To get a better understanding of the encoding complexity in H.264, let us give an overview description of the allowed modes in the H.264 state-of-the-art video encoder [56], which is illustrated in Figure 5. The encoder operates on a per-block-basis. First, block correlation to locally near and already encoded and decoded data is exploited with one of the available prediction modes using DPCM principles. Then, the remaining correlation in the prediction residual is exploited using a linear DCT-like transform, prior to quantization of transform coefficients.
Figure 5: Block diagram of the hybrid video coder architecture. The input to the coder are blocks. Block redundancies are first exploited by means of DPCM prediction with respect to reconstructed blocks in the buffer, followed by the DCT transform and subsequent quantization. Prediction parameters, such as modes and motion vectors, and transform coefficient quantization indices are encoded using a variable length code.
Lossless coding is applied to prediction parameters as well as the quantization indices of the residual transform coefficients.

Consider the prediction of a block $x_t(r,c)$ based on buffered, already encoded and decoded data, either from the spatial or the temporal buffer. Spatial prediction results in block pixels-values that are a deterministic function of pixel-values of spatially near blocks

$$x_t(r,c) = \varphi_{sm}(x_t(r-B, c), x_t(r-B, c+B), x_t(r, c-B)), \quad (31)$$

where the subscript $sm$ denotes one of the available spatial prediction modes. Temporal prediction results in block pixels-values that are a deterministic function of pixel-values of temporally near blocks

$$x_t(r,c) = \varphi_{tm}(x_{t-\Delta_t}(r-\Delta_r, c+\Delta_c), \ldots), \quad (32)$$

where the subscript $tm$ denotes one of the available temporal prediction modes, $(\Delta_t, \Delta_r, \Delta_c)$ is the motion-vector that models pixel movement from the frame $x_{t-\Delta_t}$ to frame $x_t$, and the number of arguments in $\varphi_{tm}$ depends on the prediction mode.

The H.264 standard uses square macro-blocks with side-length $B = 16$. Spatially predicted macro-blocks can be predicted as a whole, or further divided into $4 \times 4$ blocks that are predicted independently. There are four $16 \times 16$ prediction modes and nine $4 \times 4$ prediction modes, giving $4 + 9^{16}$ prediction modes in total [56]. Optimal macro-block encoding then requires a search through these prediction modes, which in turn requires for the evaluation (or at least the estimation) of the Lagrangian cost for each possible mode. Temporal prediction of macro-blocks also requires an expensive search for the optimal prediction mode. Namely, each temporally predicted macro-block is predicted on a $16 \times 16$, $16 \times 8$, $8 \times 16$, or $8 \times 8$ basis, where each $8 \times 8$ block can be in one out of four modes. In total, there are $3 + 4^4$ prediction modes, each requiring the estimation of a motion vector from at least one out of a number of available reference frames and a search area that spans the spatial neighborhood of the block in that frame. While there exists a number of fast motion estimation algorithms based on assumptions of motion vector regularity, e.g., [17, 37, 58, 61, 98, 123, 145], motion estimation for all possible modes in even one reference frame requires a computational complexity that is not available in real-time encoding on mobile devices [89, 116].

2.2 Outsourcing of Encoding Complexity

The shift to mobile devices as the main communication medium has motivated researchers to rethink the design of current video coding standards. The additional design criterion of low encoding complexity is the main
change with respect to earlier schools of thought. In particular, much of the research has been focused on eliminating or severely limiting motion estimation at the encoder, with the motivation that motion estimation in most implementations constitutes at the order of 50% of encoding complexity [49,89,97]. In this section, we describe low-complexity video encoding methods based on the idea of exploiting a feedback channel to outsource the encoding complexity to the network. While these methods are popular in the literature, we argue that they are not directly applicable to the application of real-time video communications.

Network-driven motion estimation was proposed by Rabiner and Chandrakasan in [100]. The idea was to keep the hybrid video coding structure and use a feedback channel to outsource the complexity of motion estimation to a network node or an end-node that has significantly more computational resources at disposal. The decoder would then perform motion prediction for the frame that is currently encoded using the two previously decoded frames and transmit its results to the encoder. In particular, Rabiner and Chandrakasan suggested that the decoder simply estimates the motion vectors between the two previously decoded frames and transmits them to the encoder. The encoder can use these stale motion vectors for temporal prediction, possibly after a fast refinement search. The approach was motivated by temporal coherence of the motion field [100,120] and the assumption of constant motion for sufficiently high frame-sampling rates.

A similar idea, under the name distributed video coding, was proposed by Li et al. in [71]. The idea of network-driven motion estimation was further explored by Oh et al. in [89]. The authors suggested that the motion vectors used in feedback could be improved upon if the decoder had access to a description of the current frame. Obviously, transmitting the encoded frame defies the purpose of feedback. Instead, the authors used a one-bit transform representation of the current video frame that could be used for direct estimation of the current motion vector field.

An approach to low-complexity that is considerably different from network-driven motion estimation is that of video coding based on the principles of distributed source coding. Distributed source coding refers to the problem of the encoding of a source when the decoder has access to a second (correlated) source, whose statistics are known to the encoder but not its realizations. Surprisingly, if the decoder has access to the second source, the side information, joint decoding can succeed at a rate-distortion point that corresponds to conventional conditional encoding, where also the encoder has access to the side information. Slepian and Wolf showed in [115] that this is possible for lossless encoding in case an arbitrarily small, but in general non-zero, probability of decoding error is acceptable. This result was extended to a weaker form for lossy encoding by Wyner and Ziv in [128], who showed that lossy distributed source coding in general has a non-zero loss compared to conventional coding.
Figure 6: Block diagram of the hybrid video coder architecture with network-driven motion estimation, as proposed by Rabiner and Chandrakasan [100]. Stale prediction refers to the motion estimation using the most recently decoded frames. The prediction blocks at encoder and decoder perform less computations, as they are seeded by the results from the stale prediction.
In practice, distributed source coding is often realized by means of channel codes, e.g., [1]. Consider the simple example of two correlated binary sequences \( x_1 \) and \( x_2 \) and their lossless encoding. It is possible to imagine a virtual channel that is responsible for the differences between \( x_1 \) and \( x_2 \). To encode the sequence \( x_1 \), a systematic code could be applied to \( x_1 \). The channel codeword of the systematic code consists of \( x_1 \) itself, concatenated with a sequence of parity bits. Transmission of the parity bit sequence only would, under the assumption of sufficient parity bit-rate, allow the decoder to correct the errors in \( x_2 \) with respect to \( x_1 \), and, thus, recover \( x_1 \).

Distributed source coding principles can be applied to low-complexity video encoding by considering two consecutive frames as correlated sources. This approach is commonly referred to as distributed video coding. Encoding of the current frame without reference to the previous frame removes the burden of explicit exploitation of temporal redundancy. This burden is left to the decoder. We note that the results of Slepian-Wolf and Wyner-Ziv do not apply to distributed video coding as the statistics governing frame dependency in practice are not known at the encoder, nor at the decoder.

Distributed video coding is performed in the transform domain of macro-blocks, where all coefficients of a specific band are encoded jointly [7]. This can be done by bit-plane coding using multilevel coset codes to code each bit-plane with the rate that is necessary conditioned on previously coded bit-planes [81,129]. The performance depends on the ability to model the correlation between the two frames and the efficiency of the channel code. Commonly quoted papers on the application of distributed source coding principles to low-complexity video encoding are [2, 7, 41, 99, 124]. Puri et al. applied channel coding on the quantized indices of DCT coefficients for each macro-block independently, based on statistics that were learned offline [99]. This approach has the advantage of allowing testing of many side-information candidates for each macro-block, comparable to regular motion estimation. However, the short block length of the channel code results in poor channel coding efficiency and offline learning of statistics results in poor adaptivity to the temporal correlation of the source. In [2], channel coding was applied across macro-blocks and a feedback channel was used to indicate decoding success and failure, thus improving encoder adaptivity to the source. The application of channel codes across macro-blocks allows for the use of channel codes with higher efficiency, but also results in the use of side-information with worse quality.

A combination of the network-drive motion estimation and video coding based on distributed source coding principles was proposed in [132]. The idea was to let the decoder estimate stale motion vectors, which the encoder would use to classify the macro-blocks into two classes, based on the decoder-side motion-vector predictability. The first class would be handled by regular motion compensation, while the second class would be encoded using a Wyner-Ziv coder and possibly more rounds of feedback.
Figure 7: A commonly used distributed video coding system structure, e.g., [7, 41]. The feedback channel is used to indicate decoding success and failure, allowing the encoder to transmit additional bits. The alternative is overestimating the encoding rate and settling for worse compression efficiency.
2.3 Complexity Outsourcing in Real-Time Communications

Real-time video communications have a strict low-latency requirement [85]. For this reason, the reviewed methods for complexity outsourcing by means of feedback cannot be applied to the application of real-time communication in a straightforward manner. Encoding in real-time constrains the design of feedback exploitation schemes, and it is even questionable whether complexity outsourcing by means of feedback is beneficial at all, as the feedback incurs additional power-usage [121, 130, 143].

Consider a real-time video communication system in which the decoder has just reconstructed all frames until and including frame $x_{t-1}$. Using these frames, the decoder can compute the desired model parameter, whether it be the motion vector field or a negative acknowledgement. The stale model parameters are to be transmitted over the feedback channel to the encoder, such that inter-frame redundancy may be exploited. Now consider the time instance at which the feedback data arrives at the encoder. Disregarding all computational processing delays at encoder and decoder, the encoder receives the feedback data in time for encoding frame $x_{t+[p]-1}$ for a channel with round-trip-time $p$ units of the frame sampling period. If $p$ is large and the video is dynamic, it is obvious that the estimate of motion parameters may not be as useful as when $p$ is small [65]. Therefore, it is important to design video encoders that outsource complexity via a feedback channel with the channel round-trip-time in mind.

Our main contribution to the low-complexity video encoding literature is the solution we propose to handle the requirements of real-time video communications. In particular, we use in our work the strict constraint that the use of feedback must not incur an increase in receiver playout delay. We use this constraint in Paper A to show that complexity outsourcing via feedback for Wyner-Ziv coders must be redesigned for real-time applications. We propose a simple model which naturally models the channel round-trip-time, such that the benefits from feedback can be quantified and its use optimized online. In Paper B, we investigate the design of network-driven motion estimation for channels of arbitrary round-trip-times. We use the criterion of minimization of the application-specific power usage [25, 69, 142] where we incorporate receiving power usage due to feedback. Again, the constraint on receiver playout delay is used to adjust the encoding algorithm such that complexity is outsourced only when this is beneficial from the application-specific power objective.
3 Video Distribution to Heterogeneous Receivers

Distribution of content from one source to many receivers on a packet-switched network is a problem that is commonly faced. To ensure best usage of the physical network capacity, algorithms for bandwidth efficient content distribution are sought. The importance of these algorithms comes to prominence as the number of receivers grows, and as the source data rate grows. Thus, algorithms for bandwidth efficient distribution of popular video content are of particular importance. In this section, we give a review of the theory on data distribution to one, all, and a subset of the nodes on the network. These scenarios are respectively called unicast, broadcast, and multicast. Then, two approaches for the efficient distribution of data over packet-switched networks are described. We show how these approaches have previously been extended to the case of heterogeneous receiver requirements. This provides the necessary background for Paper C and Paper D, in which we propose methods for analysis and online optimization of bandwidth efficient video distribution systems.

3.1 One-to-Many Data Distribution in Theory

Consider the transmission of source data from one sender to many receivers over a packet-switched network. The network is modeled as a directed graph $G = (V, E)$, with the set of nodes (vertices) $V$ and the set of edges $E \in V \times V$. The sender is represented by a node $s \in V$ and the receivers form the set $T \subseteq V$. The network edges are weighted and the weight $c(i, j)$ of an edge $(i, j) \in E$ is its capacity, a bound on the rate at which data may me transmitted over that edge. We are interested in maximizing the rate at which the content of source $X$ is transmitted successfully from sender $s$ to all receivers $T$.

Let us define a flow $F = \{f_{i,j} : (i, j) \in E\}$ that is the set of rates that are transmitted over the edges in the network, such that the rate on any edge does not exceed the edge capacity,

$$0 \leq f_{i,j} \leq c(i, j), \quad (33)$$

and the flow into a network node equals the flow out of that node

$$\sum_{i : (i', i) \in E} f_{i', i} = \sum_{j : (j, i) \in E} f_{i, j}, \quad (34)$$

with the exception of the source node $s$ and the receivers $T$. The value of a flow is given by

$$\sum_{(s,i) \in E} f_{s,i} - \sum_{(j,s) \in E} f_{j,s} \quad (35)$$
and represents the rate at which the source node produces data. The maximum flow of a network is the flow that has a higher value than any other flow.

Let us begin with the case of a single receiver \( t \in T \), \(|T| = 1\). Any such receiver can be disconnected from the source node \( s \in V \) by the removal of a non-unique set \( C(s, t) \subseteq E \) of edges from the graph. Such a set is called a cut, also an s-t cut, and has the capacity that equals the sum of the weight of its edges, which can be expressed as

\[
\sum_{(i,j) \in C(s,t)} c(i,j).
\]

(36)

The maximum throughput from a source to the receiver, let us call it the unicast capacity, is then clearly upper-bounded by the capacity of any such cut. In particular, the unicast capacity must be smaller than the capacity of the cut with minimum capacity, the so called minimum cut \( C_{\text{min}}(s, t) \). Menger proved in [84]—as quoted by [127]—that the capacity of the minimum cut equals the value of the maximum flow on the network. The problem of finding the minimum cut can be posed as a linear program, and its solution can be found using the simplex algorithm. Ford and Fulkerson also proved that unicast at the capacity of the minimum cut is possible, and provided a simple algorithm for finding a minimum cut [32]. The algorithm is based on finding paths from the sender to the receiver, and purging these paths from the network until no such paths remain. The capacity of the purged paths gives the capacity of the minimum cut. The complexity of the Ford-Fulkerson algorithm is proportional to \( O(|E| \sum_{(i,j) \in C_{\text{min}}(s,t)} c(i,j) / \min_{(i,j) \in E} c(i,j)) \), as the search for any path is linear in \(|E|\).

When the source content is to be transmitted to all other nodes, i.e., the receivers form a set \( T = V \setminus s \), the network resources are shared between the receivers. Transmission of physical commodities over the network then possibly has the effect of decreasing the throughput to any given receiver, since the flow from source to multiple receivers may have overlapping paths. However, in the case of data transmission, this is not necessarily the case. Namely, data can be manipulated in other ways than simple forwarding, for example replication and coding. Edmonds showed in [24]—as quoted by [127]—that the broadcast capacity is given by the minimum capacity of the minimum cut over all receivers in \( T \)

\[
\min_{t \in T} C_{\text{min}}(s, t).
\]

(37)

Edmonds showed that the broadcast capacity can be achieved by packing spanning trees on the network, where network nodes with several branches in a tree replicate incoming data and forward it along those branches.

In the case of an arbitrary number of receivers, \( T \subseteq V \setminus s \), Edmonds’s result does not hold. Thus, routing puts a limit on the achievable throughput.
However, Ahlswede et al. showed in [3], that the maximum throughput, the \textit{multicast capacity}, remains equal to the minimum capacity of the minimum cut over all receivers (37) if coding, that is outputting arbitrary functions of the inputs, is allowed in network nodes. The butterfly network in Figure 8 is a simple example where the multicast capacity equals 2 and is attainable only if coding is allowed. Li et al. showed in [72] that linear coding is sufficient to achieve the multicast capacity in acyclic networks.

\subsection{One-to-Many Data Distribution in Practice}

While the multicast capacity can be achieved using network coding, networks that allow for data coding at any node are not always available. For
example, the Internet is not such a network. In the Internet, all routers are guaranteed to forward data to one destination, while some routers can also replicate and forward to multiple destinations. This is not sufficient for achieving the multicast capacity. This section describes methods that are currently used to increase multicast throughput in networks that do not support network coding.

Several methods exist for achieving high throughput on packet-switched networks without coding at the network nodes. These methods are all based on the idea of limiting the number of times any packet is transmitted over the same network edge. This is either done by enabling packet replication in network nodes, as in IP Multicast, or by allowing for caching at certain network nodes, as in Content Delivery Networks and Peer-to-Peer Networks.

IP Multicast is a protocol that enables routers in replicating packets and forwarding them to several IP addresses. In particular, a standard is used \cite{30,50,51} to let receivers subscribe to the data stream of a multicast IP address in the network. Network routers propagate this request towards the node that is known to provide the content, keeping a table of all nodes which have reported their interest in the content. Then, as the sender transmits data, network routers replicate incoming packets and forward them to all routers that have subscribed to the content. This results in the distribution of data along a tree in the network. While IP Multicast allows for application scalability to a large number of receivers that all are interested in the same content at the same time, it has not seen full-scale deployment in the Internet. However, IP Multicast is used in private network for applications such as IPTV \cite{16}, where support in network routers can be enforced.

Content Delivery Network (CDN) is the name commonly used for the underlying network in services provided by companies such as Akamai Technologies, Limelight Networks, and EdgeCast Networks. Use of CDNs is today the predominant way of scaling content availability. One piece of evidence for that is the revenue of Akamai Technologies, which in the year 2009 amounted to USD 860 million \cite{113}. CDNs allow scalable distribution of content to a large number of receivers by means of a large network of caching servers that hold copies of content that is often accessed. The physical distribution of the servers brings content literally closer to the end receivers, thus decreasing the length of the path from the sender to the receiver and the load on the network. Clearly, CDNs can be seen as a variant of IP Multicast where replication is enabled in a subset of the network nodes. Choosing this subset wisely is one of the main optimization problems in CDN deployment, as the deployment cost is high \cite{5,8,87}. CDNs have the advantage of higher reliability due to full control over the network and the possibility to gain bandwidth efficiency when the receivers access the content over a longer period of time. While CDNs started out as a way of scaling the availability of static HTTP content, many HTTP based streaming solutions have emerged lately, such as Apple's HTTP Live
Streaming [94] and Microsoft’s Smooth Streaming, allowing the use of CDNs for video streaming solutions.

Peer-to-peer (P2P) networking is a technology that provides the caching benefits of CDNs and yet avoids the high deployment cost of CDNs. The main principle of P2P networks is to let receivers of content also play the role of distributors, in effect sharing the load of content distribution. If the receivers are close in a physical sense, P2P networks provide bandwidth efficiency by decreasing the length of paths along which data is transmitted. The main disadvantage of P2P networks is the lack of reliability and control over the distribution. Peers may at any point in time stop sharing their content. While this may not have severe consequences for file distribution, it can possibly lead to severe degradation of video quality in streaming applications [109]. Attempts to improve the reliability of P2P video distribution have included systems that are aided with a limited number of dedicated servers [108] and the use of helpers [138,139], which are peers that are not interested in the content but do help out in its distribution.

3.3 Heterogeneous Receiver Requirements

As seen from the previous two sections, there is a distinct gap between theory and practice for data distribution from one sender to multiple receivers over packet-switched networks. This section focuses on this gap and shows that it is even larger than shown so far, thus motivating the work in Paper C and Paper D.

Let us begin with the fact that a network in practice is shared for communication from more than one source of information. Yeung provided in [133] a simple example network for which it is necessary to code independent sources jointly to achieve an admissible rate point. The admissible coding rate region for multiple sources is not known [3]. One approach to dealing with the intricate problem of a shared network is to accept that sources in practice have to be encoded independently, and that the bandwidth that is available for the transmission of one particular data source varies with the state of the network. Methods for measurements of the available bandwidth on a network were reported in, e.g., [23,96], and it is common to model the packet-loss due to network congestion by a two-state Markov model [42,47,73,125].

The approach of independent source coding is often adopted in unicast communication, aided by traffic shaping algorithms to avoid network congestion. The Transmission Control Protocol (TCP), for example [35], is a protocol that adapts the transmission rate to the state of the network. TCP is based on the principle of end-to-end acknowledgement of received packets and retransmissions of lost packets. While TCP guarantees successful transmission, it has no guarantees on transmission rate and is not applicable to real-time applications due to long round-trip-times, especially for
communication over long physical distances. In video streaming distribution over CDNs, where communication latency is not a strict as in real-time applications, TCP has proved to be a viable approach [68, 107] as a large enough CDN ensures proximity of receivers to a CDN server.

Another problem that was not addressed in Section 3.1 nor Section 3.2 is that the multicast objective itself might be flawed. As mentioned earlier, Ahlswede et al. showed that the multicast capacity equals the smallest unicast capacity over the set of receivers [3]. Thus, the receiver with the smallest minimum cut limits the data-rate to other receivers. What if we are interested in communication with a non-zero distortion, where receivers may reconstruct the source data at different levels of fidelity? Sarshar and Wu showed in [106] that the rate transmitted to each receiver can be equal to that receivers unicast capacity if an embedded source coder, with successive refinement [29], is used and the network has a specific topology, where receivers with a smaller unicast capacity are located spatially “below” receivers with a larger unicast capacity. The unicast capacity of each receiver is achieved by encoding the source in layers, with rates that are equal to the distinct unicast capacities of the receivers, and using transmission by independent network coding for each source layer. However, successive refinement is not possible for all sources and some losses are necessary in practice.

Sarshar and Wu’s approach to adapt to the unicast capacity of each receiver is well known in networks with no network coding [122, 141]. There, multiple multicast trees are used to increase the adaptivity to the unicast capacity of each receiver. *Simulcast* is the name used to denote such transmission if the stream distributed in each multicast tree is decodable by itself, and *layered multicast* is used to denote the transmission where the stream distributed in each multicast tree is a layer of an stream encoded by an embedded coder.

The number of multicast trees is limited both in simulcast and layered multicast. Namely, simulcast streams are redundant and not very bandwidth efficient as the number of multicast trees grows. In layered multicast on the other hand, the number of layers required from an embedded coder affects the performance of the coder in general. Thus, a problem in video distribution over packet-switched networks is the optimization of the encoding rates for a limited number of streams [18, 60, 63, 64, 77, 78, 131, 135].

Our main contribution to the video distribution literature is the optimization of multi-rate encoders in bandwidth efficient video distribution technologies that utilize several multicast trees to adapt to a large number of receivers with heterogeneous requests. In particular, we provide an interpretation of the simulcast and layered multicast problems that allows for the analytical optimization of the rates of the streams transmitted over the different multicast trees. Our work is applicable to IP Multicast as well as CDNs.
4 Summary of Contributions

The focus of this thesis is on the development of methods for efficient video communication over packet-switched networks. The main contributions of the thesis can be summarized as

- improvements in the modeling and solution of the low-complexity video encoding problem for real-time communication over packet-switched networks with an available feedback channel, and

- novel interpretation of the multi-rate video encoding problem in content delivery networks and multicast systems, leading to the straightforward application of methods from quantization theory towards optimal multi-rate video encoding.

The thesis consists of four research papers, in which I formulated the addressed problems and the approaches to their solutions, as well as the mathematical derivations and the experimental validation. The co-authors provided important knowledge and know-how in the topics of distributed source coding and quantization, as well as a forum for discussion of the state-of-the-art. Short summaries of the papers are presented below.

Paper A: Frame-Bufferless Sum-Rate Constrained Video Encoding using Feedback

In the context of low-complexity video encoding on mobile devices, we investigate the popular idea of complexity outsourcing to a network node by means of a feedback channel and the use of distributed source coding principles. In particular, we focus on the real-time video communication application, which highlights arguments both for and against the use of feedback. These are, respectively, the low-complexity requirement, the low-latency requirement, the availability of a feedback channel, and the changing state of the video characteristics and the communication channel. We propose a simple online statistical model that explores how to best utilize the feedback and distributed source coding mechanisms to minimize the overall communication rate, under strict constraints on receiver playback delay. Our analysis and experimental results validate the efficiency of the proposed system.

Paper B: Power-Constrained Low-Latency Video Encoding using Feedback

In Paper B, we question and abandon the use of distributed source coding for low-complexity video encoding for real-time video communication. Instead of setting strict constraints on the structure of the video encoder—such as the requirement of no motion estimation—we adopt the broader criterion
of application-specific power consumption, comprising the total power of video encoding and transmission. We introduce a general framework based on a novel sum-rate-criterion that allows for convenient abstraction of implementation specific parameters in encoder design. With the help of the proposed framework, we design a feedback enabled H.264 compatible encoder, which gives substantial improvements over state-of-the-art low-complexity video encoders in terms of both application-specific power consumption and weighted-sum-rate.

**Paper C: A Quantization Theoretic Perspective on Simulcast and Layered Multicast Optimization**

The topic in Paper C is that of bandwidth efficient video distribution to a large number of receivers over erasure-free networks. We consider the rate optimization problem in multicast systems that use multiple trees to accommodate heterogeneous available bandwidths of receivers in the network. For specific network topologies, we show that the multicast rate optimization problem is equivalent to the optimization of scalar quantizers. With this contribution, we make available all the tools that exist in the well explored field of quantizer design. For example, we use results from rate-distortion theory to provide a bound on the achievable performance for the multicast rate optimization problem. Further, we derive an analytic solution to the problem that is asymptotically optimal in the number of multicast trees based on the high-rate approximation. The analytic solution is especially valuable in dynamic networks, such as the Internet. We also derive local optimality conditions and a general class of iterative algorithms that give locally optimal solution to the problem.

**Paper D: On Bandwidth-Efficient Video Distribution Through Multi-Rate Video Encoding**

In Paper D, we continue with the problem of video distribution. Using our results from Paper C as a starting point, we extend our model to include packet-loss events. Based on the quantization theoretic interpretation of the problem, we investigate the performance of two bandwidth efficient video distribution technologies, Content Delivery Networks (CDNs) and IP Multicast, which use multi-rate video encoding as means to strike a balance between usage of the network bandwidth and heterogeneous receiver-demands. We provide analytic solutions to the encoding-rate optimization problem for both CDNs and IP Multicast. Comparison between the closed-form expressions of the expected receiver distortion for the two video distribution technologies quantifies the cost of the low-latency requirement that is inherent in IP Multicast.
References


