

MPEG-4, H.264 and H.265 Video Bandwidth Prediction via Markovian Models and Simulated Annealing



Athina Kalampogia

Thesis Advisor: Associate Professor Polychronis Koutsakis

Committee member: Associate Professor Michail Lagoudakis

Committee member: Professor Michael Paterakis

School of Electrical and Computer Engineering

Technical University of Crete

This thesis is submitted for the degree of:

Master of Science

March 2017

Acknowledgements

I would like to thank, for the second time, my advisor Polychronis Koutsakis not only for giving me the opportunity to work with him in my Master's thesis but for all the guidance and support from the beginning of the thesis until the end. Moreover, I would like to thank Prof. Michael Paterakis for his help in proper conduct of my scholarship procedure and Prof. Michail Lagoudakis for his participation in my committee.

I would also like to express my gratitude to Giannis Klasinas for his help and the long discussions about data analysis techniques.

Nothing would be the same without my dear friends, by my side to support me and help me overcome the difficulties I have faced. I would like to deeply thank my family for their unconditional love and support. Last but definitely not least, a special "Thank you" to Manolis Kousanakis for always believing in me and making me believe in myself, giving me the boost we all need sometimes.

Abstract

The explosive growth of multimedia applications renders the efficiency of network resource allocation a problem of major importance. The burstiness of video traffic, in particular, calls for traffic control solutions that will help prevent significant packet losses. Such losses can lead to unacceptable Quality of Service (QoS) and Quality of Experience (QoE) to users. In this work, we focus on a large variety of MPEG-4, H.264 and H.265-encoded video traces with different structural patterns. Different versions of each trace, in low, medium and high quality have been used in our study. We evaluated the accuracy of an existing video traffic prediction approach for the size of B-frames and tested some variants of it. We implemented the metaheuristic technique of Simulated Annealing to predict the size of B-frames, and compared the new results against an existing approach from the literature. We propose a new Markovian model that predicts B-frames' sizes with significantly higher accuracy. B-frame size prediction can be used in order to reduce bandwidth requirements and smoothen the encoded video stream, by selective B-frame dropping, when the model predicts larger upcoming B-frame traffic than the network can handle.

Table of contents

List of figures	ix
List of tables	xi
Acronyms/Abbreviations	xiii
1 Introduction	1
1.1 Network Traffic	1
1.1.1 Video compression standards	1
1.2 Motivation	2
1.3 Thesis Contribution	3
1.3.1 Publications	4
1.4 Thesis Outline	4
2 Background	5
2.1 Related Work	5
2.2 Video Traffic Characteristics	6
2.2.1 Coefficient of Variation	7
2.2.2 Size Comparison of Frames	9
3 Prediction by Exploiting Inter-Frame Correlation Coefficients	11
3.1 Introduction	11
3.1.1 Coefficient of Correlation	11
3.1.2 Autocorrelation of B-frames	13
3.2 Predicting the size of B-frames	14
3.2.1 Implementation	14
3.2.2 Results and Comparison	16

4	Simulated Annealing	19
4.1	Indroduction	19
4.2	Using Simulated Annealing for prediction	20
4.2.1	Implementation	20
4.2.2	Results and Complexity	21
5	Markovian Model	25
5.1	Outliers	25
5.2	Big and small B-Frames	28
5.3	The Proposed Model	31
5.4	Results	38
5.5	Comparison	43
5.5.1	With LSTAR Model	43
5.5.2	With Markov-modulated Gamma-based model	45
6	Conclusion and Future Work	49
	References	51

List of figures

1.1	Frame interaction	2
4.1	Example of a local vs absolute maximum	19
5.1	Markov chain for prediction of the B1-frames for “Lake House” video trace in QP=10	36
5.2	Markovian Model – QQ plot results for the Harry Potter QP=10 and QP=35 traces	44
5.3	Markovian Model – QQ plot results for the NBC news QP=10 and QP=35 traces	45

List of tables

2.1	CoV Results for H.264 Video Traces	7
2.2	CoV Results for H.265 Video Traces	8
2.3	I- and B-frames Size Comparison for H.264 Star Wars IV Trace	9
2.4	I- and B-frames Size Comparison for H.265 Traces	10
3.1	Correlation Coefficient for H.264 Star Wars IV Trace	12
3.2	Correlation Coefficient for H.265 Lake House trace	12
3.3	IFCC-based model - RPE Results for H.264 Video Traces	17
3.4	IFCC-based model - RPE Results for H.265 Video Traces	18
4.1	RPE Results for MPEG-4 Video Traces	22
4.3	RPE RESULTS FOR H.265 VIDEO TRACES	23
4.2	RPE Results for H.264 Video Traces	24
5.1	RPE for prediction of “small” B-frames’ sizes for H.264 videos	26
5.2	RPE for prediction of “small” B-frames’ sizes for H.265 videos	27
5.3	RPE results for H.265 video traces using BBFs and SBFs	29
5.4	RPE results for H.264 video traces using BBFs and SBFs	30
5.5	Best Fitting distributions for big and small B-frames After 50 repetitions for NBC News $G_{16}B_1$ QP=28	34
5.6	Best Fitting distributions for big and small B-frames for H.264 Video traces	35
5.7	Best Fitting distributions for big and small B-frames for H.265 Video traces	36
5.8	Markov Model- RPE results for H.265 Video Traces	39
5.9	Markov Model- RPE results for H.264 Video Traces	40
5.10	RPE results after ”Repeated random sub-sampling validation” for H.265 Video Traces	41

5.11 RPE results after “Repeated random sub-sampling validation” for H.264 Video Traces	42
5.12 Markov Model- RPE results for MPEG-4 Video Traces	43
5.13 NMSE Comparison with [1]	44
5.14 RPE comparison with [2] for H.265 Video Traces	46
5.15 RPE comparison with [2] for H.264 Video Traces	48

Acronyms/Abbreviations

BBF Big *B*-frame

CoV Coefficient of Variation

GoP Group of Pictures

IFCC Inter-Frame Correlation Coefficient

LRD Long Range Dependence

MAPE Mean Absolute Percentage Error

NMSE Normalized Mean Square Error

QoE Quality of Experience

QoS Quality of Service

RPE Relative Percentage Error

SBF Small *B*-frame

SRD Short Range Dependence

Chapter 1

Introduction

1.1 Network Traffic

According to the latest Cisco estimates, by 2020 a million minutes of video content will cross global IP networks every second. IP video traffic will correspond to 82% of all consumer Internet traffic. It is already more than 70%, currently [3].

As video services occupy an overwhelming percentage of the total network traffic in wired and wireless networks, it becomes all the more necessary to optimize the way in which bandwidth is allocated. Providers are faced with a significant problem: if they choose to reduce the bandwidth offered for a user's transmission (in order to allow more users to enter the network) there is the possibility of losing packets that are crucial for the quality of the video; on the other hand, if they constantly overallocate bandwidth to handle possible video traffic bursts, this will lead to bandwidth wastage.

1.1.1 Video compression standards

In this work, we focus on videos encoded with the MPEG-4, H.264 and H.265/High Efficiency Video Coding (HEVC) compression standards. H.264 is currently one of the most commonly used video coding standards as it covers a huge range of applications like videoconferencing, mobile services and HD video storage [4]. H.265 [5] is its successor, and its goal is to achieve the same video quality at lower bit rates. There are two basic structural features that common MPEG-4, H.264 and H.265 encoders share: the three types of frames (I , P , B) they generate and the pattern in which these frames are generated.

I -frames are completely self-referential and don't use information from any other frames. They provide a point of access to the compressed video data. I -frames are

the largest among the 3 types of frames but the least efficient from a compression perspective. P-frames are “predicted” frames. When producing a P-frame, the encoder can look backwards to previous I or P-frames for redundant picture information. P-frames are smaller than the *I*-frames. *B*-frames are bi-directionally predicted frames. This means that when producing *B*-frames, the encoder can look both forwards and backwards for redundant picture information. The encoders use a fixed Group-of-Pictures (GoP) pattern when compressing a video sequence. There are two variables that define this pattern: the distance between *I*-frames and the distance between P-frames. The values of those variables vary depending on the required video quality and the transmission rate. The way in which the frames of a GoP use information from preceding or successive frames for encoding is shown in Figure 1.1.

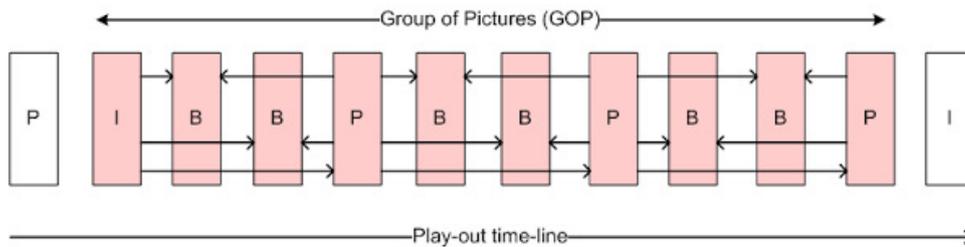


Fig. 1.1 Frame interaction

MPEG, H.264 and H.265 encoders achieve compression by making small quality compromises in ways that are intended to be minimally perceptible [6]. In particular, the quantization parameter QP regulates how much spatial detail is saved. When QP is very small, almost all that detail is retained. As QP is increased, some of that detail is aggregated so that the bit rate drops – but at the price of some increase in distortion and some loss of quality.

1.2 Motivation

The subject of video traffic modeling has been widely studied in the literature. An accurate model of the expected video traffic will allow network administrators to make better estimates of the bandwidth that is required for the Quality of Service(QoS) and Quality of Experience(QoE) of their customers. For example, video chatting on mobile devices demands high bandwidth and if the offered bandwidth is decreased video quality may be compromised [7]. Also, regardless of the network, it is inefficient to use the same representation of a video for the duration of a streaming session. Instead,

it must be adapted to dynamically varying networks conditions such as throughput, packet loss rate, and delay jitter [8]. These problems can be significantly mitigated if the volume of video traffic that will be generated in the network can be predicted. Such a model can, e.g., be very useful in determining the proper admission control algorithm to ensure that no degradation takes place in the QoE of accepted video sessions [9].

An admission control algorithm does not need to guarantee the transmission of all video packets. On the contrary, in order to ensure that video traffic does not use all the bandwidth in the link and cause other important traffic to experience dropped packets [10], an efficient admission control algorithm must take into account that in video traffic, certain packets can be dropped if the network is congested, with controllable decoding quality degradation. Hence, an efficient scheme should smartly drop frames or packets when the communication links are congested or bandwidth is limited [11].

Usually, in an MPEG-4, H.264 or H.265 GoP there are more *B*-frames than *P* or *I*-frames. This leads to *B*-frames occupying the largest percentage of bandwidth. Comparing the three types of frames and the problem that the loss of each can cause, the loss of *B*-frames is less harmful than the loss of any other type of frame [12]. This is due to the fact that *B*-frames use only the differences between the current frame and both the previous and following frames to specify their content, so their loss causes motion artifacts that the human eye has difficulty to understand unless there is a high loss rate. On the other hand, the loss of *I*-frames or *P*-frames can cause image distortion that is observable even in the case of low loss rate.

Therefore, in order to reduce bit rate variability and have a smoother video bit stream it would be very beneficial for network administrators to be able to predict the volume of traffic associated with *B*-frames and, in the case of network congestion, that traffic (or portion of it) could be dropped. The possible dropping of *B*-frames can also be used for scheduling purposes, to improve bandwidth utilization [13].

1.3 Thesis Contribution

Our work focuses on the prediction of the size of the *B*-frames of the GoP for H.264, H.265 and MPEG-4 video traces. In [14], real time algorithms are proposed to predict the size of *B*-frames for MPEG-4 video traffic. In our previous work [15] we tried to confirm whether the same methodology also works well for H.264 videos and tested a new model that we have designed. In this Master thesis following the same procedure as in [15] we applied these methods for H.265 video traces. As it will be shown in the following Sections, we found that the approach used in [14] does not provide good

results neither in the case of H.264 videos nor for H.265 videos. Furthermore, we have implemented Simulated Annealing for predicting B -frames sizes, for MPEG-4, H.264 and H.265 video traces and we have found that the new technique excels in comparison to the proposed approach in [14] for all three encodings used in our study. Utilizing the conclusions we derived on the reasons why the approach of [14] does not work well for H.264 and H.265 videos, we created in this Thesis, a different model for prediction of B -frame's sizes by separate them in two different classes. We proceeded to propose and implement a Markovian model using multiple distributions which is shown to achieve high accuracy with low complexity, for all the video traces used in our study. These results are especially important since these videos have large differences in content, encoding quality and GoP structure.

1.3.1 Publications

The research work conducted during this Master's thesis has led to the following publications:

- A. Kalampogia and P. Koutsakis, "Using Simulated Annealing for Improved Video Bandwidth Prediction", in Proc. of the IEEE INFOCOM 2017 Workshop on Communication and Networking Techniques for Contemporary Video
- A. Kalampogia and P. Koutsakis, "H.264 and H.265 Video Bandwidth Prediction", revised for publication in the IEEE Transactions on Multimedia

1.4 Thesis Outline

The Master's thesis is organized as follows : In Chapter 2 we present the related background with some of the basic traffic characteristics of video traces. In Chapter 3 we analyze the frame's correlation and we evaluate an existing model. We present a different variation of this model and compare the results. The use of Simulated Annealing is described in Chapter 4. In Chapter 5, we present the proposed Markovian Model along with its evaluation and comparison with other similar works. Finally, we conclude this thesis in Chapter 5 with the discussion of ideas for future research.

Chapter 2

Background

As analyzed in [2], video models can be classified into two categories: (a) data-rate models, and (b) frame-size models. In a data-rate model, only the rate at which data arrive at a link is generated for performance prediction purposes. Most models fall under this category. These models generally achieve good results in predicting average packet-loss probability and buffer overflowing probability. However, they have the shortcoming of failing to identify such details as the percentage of frames affected, as even a small rate of data loss involving I frames may affect the perceptual quality of the received video significantly, but the same amount of data loss in B frames would have far smaller impact. In a frame-size model, sizes of individual video frames are generated, and hence, data-rate information can be obtained from the frame-size information.

2.1 Related Work

The modeling approaches that have been used in the literature include first-order autoregressive (AR) models [16], Markov renewal processes (MRP) [17], finite-state Markov chain [18, 19], wavelet modeling [20], recurrent neural networks [21], seasonal Autoregressive Integrated Moving Average (ARIMA) [22, 23], and Gamma-beta-autoregression (GBAR) models [24, 25]. In [26, 27] the authors show that H.261 video-conference sequences generated by different hardware coders, using different coding algorithms, have gamma marginal distributions (this result was also employed by [28]) and use this result to build a Discrete Autoregressive (DAR) model of order one, which works well when several sources are multiplexed. Similarly, in [29–31] the authors show that the traffic generated from H.263 and MPEG-4 videoconference sequences is better approximated by the Pearson V distribution and they use this result to build DAR models.

In [32] the authors compute the probability of timely delivery of streamed video packets over an LTE network as a function of the bandwidth allocated to the user, without differentiating between the importance of the delivery of individual video frame types. However, [32] follows the directions of the DASH standard [33], according to which several copies of the same video need to be encoded at different rates and stored on a content server; as efficient as DASH is showing to be, it is not adapted for very low latency streaming such as videoconferencing or interactive applications, such as live webcasting [34]. Additionally, even in the case of using the adaptive bitrate streaming technology of DASH for content that does not require very low latency streaming, the various copies of the same video still include frames of lower importance. These frames can be dropped when the network is congested.

2.2 Video Traffic Characteristics

It is necessary to understand how the size of the different types of frames in a GoP affect the total bandwidth requirements and the size of impact their loss would have in the quality of the received video.

As mentioned before the encoders use a fixed Group-of-Pictures (GoP) pattern when compressing a video sequence. The GoP structure for MPEG-4, H.264 and H.265 videos is defined as G_aB_z , where a is the total number of frames in the GoP and z is the number of consecutive B-frames in each GoP. In our research we used several video traces for MPEG-4, H.264 and H.265 compression standards, obtained from [35–37]. More specifically, for H.264 we used the video traces of “Silence of the Lambs”, “NBC-news” and “Star Wars IV” each of them in 4 different types of GoP ($G_{16}B_1$, $G_{16}B_3$, $G_{16}B_7$ and $G_{16}B_{15}$) and each of them in Low Quality (LQ), with QP=48, Medium Quality (MQ) with QP=28 and High Quality (HQ) with QP=10. Similarly for H.265, we worked with 24 different video traces. We used the video traces of “Big Buck Bunny”, “Elephants Dream”, “Tears of Steel”, “Blue Planet”, “Lake House”, “Speed”, “Finding Neverland” and “Harry Potter” each of them in Low Quality (LQ), with QP=40, Medium Quality (MQ) with QP=25 and High Quality (HQ) with QP=10. All the traces use the compression format $G_{24}B_7$. For the MPEG-4 compression standard we used the video traces of “Silence of the Lambs” in HQ, MQ and LQ, “Aladdin” in HQ, MQ and LQ, “Ski” in HQ and Simpsons in HQ. The words “trace” and “movie” are used interchangeably throughout the rest of the thesis.

2.2.1 Coefficient of Variation

In [14] the coefficient of variation (CoV) of the frame sizes was used for MPEG-4 traces to represent the rate variability. Likewise, we use CoV of the frame sizes to represent the rate variability for H.264 and H.265 video traces. For a video sequence consisting of M frames encoded with a given quantization level, if $X_m (m=1,2,\dots,M)$ denotes the sizes of the encoded video frames, then the CoV of the encoded video is defined in equation 2.1, where σ is the standard deviation and \bar{X} is the mean of the frame sizes.

$$CoV = \frac{\sigma}{\bar{X}} = \frac{\sqrt{\frac{1}{(M-1)} \sum_{m=1}^M (X_m - \frac{1}{M} \sum_{m=1}^M X_m)^2}}{\frac{1}{M} \sum_{m=1}^M X_m} \quad (2.1)$$

In order to observe how B -frames affect the rate variability of each movie we computed the CoV of the frame sizes of the trace with and without B -frames. Hence, we initially computed the CoV of the frame sizes for the whole trace. Then, we removed the B -frames, and computed the new CoV (mean and standard deviation) using only the I- and P-frames. The results for H.264 videos are shown in Table 2.1 and for the H.265 video traces in Table 2.2.

Table 2.1
CoV Results for H.264 Video Traces

QP	$G_{16}B_1$		$G_{16}B_3$		$G_{16}B_7$		$G_{16}B_{15}$	
	Cov with B- frames	CoV without B- frames						
Silence of the Lambs								
10	1.012	0.678	1.133	0.614	1.070	0.552	0.974	0.487
28	2.334	1.606	2.344	1.329	2.271	1.065	2.068	0.838
48	1.834	1.439	1.988	1.863	2.078	0.920	2.091	0.736
NBC-News								
10	0.337	0.236	0.335	0.243	0.330	0.239	0.307	0.225
28	1.372	0.954	1.478	0.806	1.433	0.622	1.285	0.417
48	1.854	1.286	2.064	1.005	2.160	0.694	2.163	0.402
Star Wars IV								
10	0.995	0.640	1.054	0.556	0.992	0.457	0.865	0.329
28	1.745	1.232	1.834	0.988	1.761	0.740	1.574	0.491
48	1.709	1.321	1.814	1.047	1.811	0.743	1.719	0.466

It is easy to observe from the results that in all cases the coefficient of variation (CoV) is significantly reduced when B -frames are removed from the entire encoded

Table 2.2
CoV Results for H.265 Video Traces

	CoV with B-frames	CoV without B-frames
Big Buck Bunny QP=10	1.717	0.786
Big Buck Bunny QP=25	2.981	1.169
Big Buck Bunny QP=40	3.873	1.437
Blue Planet QP=10	1.116	0.572
Blue Planet QP=25	2.950	1.086
Blue Planet QP=40	4.426	1.538
Speed QP=10	0.518	0.248
Speed QP=25	1.557	0.640
Speed QP=40	1.731	0.691
Lake House QP=10	0.753	0.392
Lake House QP=25	2.577	1.064
Lake House QP=40	3.371	1.323
Finding Neverland QP=10	0.670	0.266
Finding Neverland QP=25	1.904	0.672
Finding Neverland QP=40	2.565	0.944
Elephants Dream QP=10	1.454	0.861
Elephants Dream QP=25	2.346	1.218
Elephants Dream QP=40	3.244	1.441
Tears of Steel QP=10	0.744	0.446
Tears of Steel QP=25	1.954	0.855
Tears of Steel QP=40	3.176	1.220
Harry Potter QP=10	0.590	0.289
Harry Potter QP=25	1.904	0.699
Harry Potter QP=40	2.470	0.947

video, i.e. the I and P frames have smaller variations in their sizes. Hence, the reduction of this rate variability in cases of network congestion would lead to smoothing the encoded video bit stream. Another noteworthy observation is that in all cases, high quality video traces (QP=10) exhibit lower CoV than medium and low quality movies. This is expected as the more compressed a movie is, the more likely it is to contain outliers leading to high CoV. Comparing the results for the two different compression standards we can see that in the case of H.265 video traces and especially in low and medium quality, the CoV results are higher than those for H.264 videos. The higher bit rate variability of H.265 videos calls for even higher accuracy in predicting video bandwidth requirements.

2.2.2 Size Comparison of Frames

In order to further consider how the B -frames sizes affect the total bandwidth requirements of a trace, we compared the size of B -frames with the size of I -frames, for the H.264 and H.265 movies used in our study. Table 2.3 presents the size comparison for the H.264 Star Wars IV movie in three different GoP patterns (the results were qualitatively similar for the other movies). Table 2.4 presents the respective results for three H.265 video traces (“Lake House”, “Blue Planet” and “Big Buck Bunny”).

Table 2.3
I- and B-frames Size Comparison for H.264 Star Wars IV Trace

QP	Total size of I-frames (bytes)	Total size of B-frames (bytes)	Mean size of I-frames (bytes)	Mean size of B-frames (bytes)	Mean aggregate size of the B-frames per GoP (bytes)
Star Wars IV $G_{16}B_1$					
10	7,68E+07	7,74E+07	2,28E+04	2,87E+03	2,29E+04
28	1,32E+07	4,27E+06	3,92E+03	1,58E+02	1,27E+03
48	1,47E+06	6,26E+05	4,34E+02	2,30E+01	1,86E+02
Star Wars IV $G_{16}B_7$					
10	7,95E+07	2,43E+08	2,35E+04	5,15E+03	7,21E+04
28	1,38E+07	1,76E+07	4,08E+03	3,72E+02	5,20E+03
48	1,50E+06	1,60E+06	4,44E+02	3,40E+01	4,75E+02
Star Wars IV $G_{16}B_{15}$					
10	8,16E+07	3,30E+08	2,42E+04	6,51E+03	9,77E+04
28	1,44E+07	2,72E+07	4,25E+03	5,37E+02	8,05E+03
48	1,55E+06	2,21E+06	4,58E+02	4,40E+01	6,55E+02

As expected, the mean size of I -frames in a trace is always several times larger than the mean size for B -frames. In a GoP though, there is only one I -frame and as many B -frames as the GoP structure dictates. We also observe the difference of the average size of I -frames with the mean aggregate size of B -frames per GoP. We should note that the results differ for every type of GoP because the number of B -frames per GoP changes. For example, for the movie “Star Wars IV”- $G_{16}B_1$ we can see that the average aggregate size of all B -frames per GoP is smaller than the average size of I -frames for the MQ and LQ versions of the trace, and comparable with the average size of I -frames for the HQ trace (QP=10). This is so because in $G_{16}B_1$ there are only 8 B -frames per GoP. On the other hand in, “Star Wars IV”- $G_{16}B_7$ and $G_{16}B_{15}$ the mean aggregate size of B -frames per GoP is significantly larger than the mean size

Table 2.4
I- and B-frames Size Comparison for H.265 Traces

QP	Total size of I-frames	Total size of B-frames	Mean size of I-frames	Mean size of B-frames	Mean aggregate size of the B-frames per GoP
Blue Planet $G_{24}B_7$					
10	1,11E+09	4,06E+09	2,91E+05	7,61E+04	1,60E+06
25	1,51E+08	1,13E+08	3,97E+04	2,12E+03	4,44E+04
40	2,39E+07	6,94E+06	6,26E+03	1,30E+02	2,73E+03
Lake House $G_{24}B_7$					
10	1,36E+09	6,68E+09	2,52E+05	8,84E+04	1,86E+06
28	1,54E+08	1,69E+08	2,85E+04	2,23E+03	4,69E+04
48	2,49E+07	1,71E+07	4,62E+03	2,27E+02	4,76E+03
Big Buck Bunny $G_{24}B_7$					
10	2,33E+08	4,76E+08	2,61E+05	3,81E+04	8,01E+05
28	4,90E+07	3,98E+07	5,48E+04	3,19E+03	6,70E+04
48	8,66E+06	4,22E+06	9,69E+03	3,38E+02	7,10E+03

of *I*-frames due to the fact that the number of *B*-frames increases. In all cases, a very substantial portion of every trace consists of *B*-frames; hence selective *B*-frames dropping can prove to be very efficient for network traffic control, in the case of network congestion.

Chapter 3

Prediction by Exploiting Inter-Frame Correlation Coefficients

3.1 Introduction

As explained before, B -frames are constructed based on the reference I and P -frames, hence we expect a possible strong correlation with the size of those frames. In order to achieve an accurate prediction for B -frame sizes, we have studied the correlation of B -frames' sizes with I - and P -frames' sizes. In a GoP, every B -frame has its own importance, so we must study separately each and every one's correlation with the other types of frames.

3.1.1 Coefficient of Correlation

We compute the coefficient of correlation ($P_{(X,Y)}$) between the size of each B -frame (variable X), and the size of each I - or P - frame (variable Y) in a GoP using equation 3.1.

$$P_{(X,Y)} = \frac{E(XY) - \bar{X}\bar{Y}}{\sigma_x\sigma_y} \quad (3.1)$$

where σ_x and σ_y represent the standard deviation of X and Y respectively. Indicatively, the correlation results for the H.264 "Star Wars IV" $G_{16}B_3$ MQ trace are presented in Table 3.1 and the results for the "Lake House" $G_{24}B_7$ HQ trace are presented in Table 3.2.

Table 3.2
Correlation Coefficient for H.265 Lake
House trace

Table 3.1
Correlation Coefficient for H.264 Star
Wars IV Trace

Star Wars IV $G_{16}B_3$ QP=28				
	I	P1	P2	P3
B1	0,229	0,664	0,630	0,548
B2	0,240	0,732	0,692	0,608
B3	0,226	0,703	0,657	0,579
B4	0,214	0,677	0,724	0,609
B5	0,224	0,713	0,769	0,655
B6	0,208	0,665	0,723	0,616
B7	0,205	0,625	0,670	0,671
B8	0,227	0,672	0,719	0,715
B9	0,199	0,587	0,643	0,658
B10	0,203	0,561	0,614	0,628
B11	0,212	0,621	0,652	0,660
B12	0,207	0,590	0,612	0,619

Lake House G24B7 QP=10			
	I	P1	P2
B1	0.494	0.573	0.554
B2	0.568	0.695	0.713
B3	0.498	0.589	0.544
B4	0.746	0.803	0.743
B5	0.492	0.633	0.599
B6	0.545	0.646	0.592
B7	0.452	0.579	0.569
B8	0.474	0.601	0.586
B9	0.535	0.623	0.696
B10	0.469	0.612	0.595
B11	0.699	0.762	0.810
B12	0.449	0.589	0.644
B13	0.564	0.632	0.655
B14	0.432	0.549	0.581
B15	0.476	0.597	0.602
B16	0.528	0.620	0.623
B17	0.447	0.595	0.616
B18	0.681	0.763	0.762
B19	0.438	0.545	0.591
B20	0.556	0.683	0.624
B21	0.417	0.530	0.550

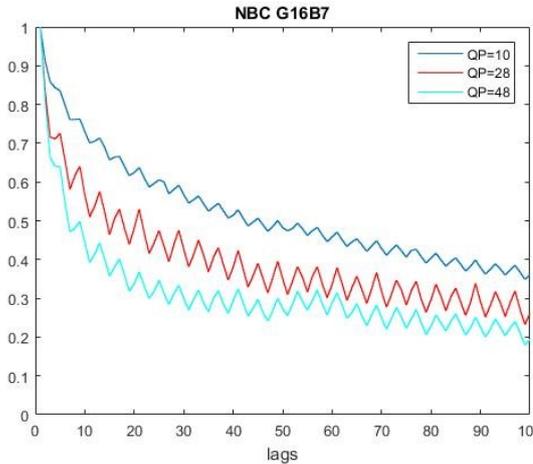
The values in bold show the strongest correlation for every B -frame. It is clear that for all cases the correlation between B -frames and I -frames is weaker than the correlation between B - and P - frames. Even when B -frames are encoded with an I -frame as reference (for instance, the first B -frame in $G_{16}B_1$), the coefficient of correlation between them and the I -frame is lower than the coefficient of correlation between them and a P -frame (except of course in the case of the $G_{16}B_{15}$ pattern, in which there are no P -frames). This was true for all H.264 and H.265 traces under study.

3.1.2 Autocorrelation of B-frames

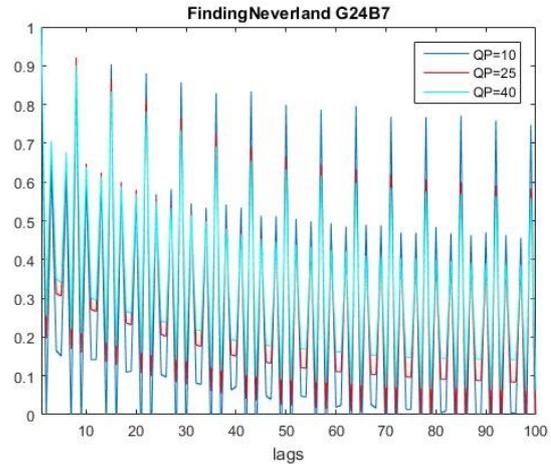
Our next step was to compute the autocorrelation between the B -frame of a movie trace. In equation 3.2 X represents the size of the B -frame, σ_x represents the standard deviation of X and k is the lag.

$$r_k = \frac{E[(X_m - \bar{X})(X_{m+k} - \bar{X})]}{\sigma_x^2} \quad (3.2)$$

We have computed the autocorrelation for every trace using lags from 1 to 1000. For the H.264 compression standard, for the majority of the traces the values of the curves were very close to 1 for the first lags, indicating a strong Short Range Dependence (SRD), however the autocorrelation of B -frames quickly decreased, indicating an absence of Long Range Dependence (LRD). As for the H.265 encoded video traces, the curves of the autocorrelation between B -frames show a different behavior. The strongest correlations are observed for the vast majority of the video traces for $lag = 7$ and $lag = 14$. The high autocorrelations for lags that are multiples of 7 can be explained by the GoP structure ($G_{24}B_7$) of the traces. Figures 3.1a and 3.1b show the above-mentioned trends, indicatively, for the H.264 “NBC-news” $G_{16}B_7$ trace and the H.265 “Finding Neverland” $G_{24}B_7$ trace.



(a) H.264
“NBC-news” $G_{16}B_7$ trace



(b) H.265
”Finding Neverland” $G_{24}B_7$ trace

3.2 Predicting the size of B-frames

3.2.1 Implementation

The high values of the autocorrelation between B -frames' sizes and the coefficient of correlation between B -frames and P -frames, led us to experiment, in my Diploma thesis [15] both with the approach used in [14] and with a variation of that approach for H.264 movie traces. In this Master thesis, we extended this work and we have implemented the above models, for H.265 video traces, in order to test if the prediction results was accurate for the new compression standard.

The authors in [14] used inter-frame correlation coefficients (IFCCs) for a linear regression-based B -frame size prediction. The methodology is as follows: for each movie, we choose among all autocorrelation values and all coefficient of correlation values the two largest ones. For the prediction of each B -frame size in the next GoP we use the two frames that have on average the highest correlation with the specific B -frame, over all GoPs of the trace. For example, if the two highest values are the autocorrelation values for lag=1 and lag=2, then the two previous B -frames' sizes will be used in the set of equations for the prediction of the next B -frame size. If, on the other hand, the correlation of the B -frame whose size we want to predict is higher with a P -frame, then that P -frame size will be used in the set of equations.

To define the size of the first B -frame of the t^{th} GoP of each movie we used the notation $B_{1,t}$. We continued, similarly, for the rest of the B - and P - frames ($B_{2,t}, B_{3,t}, B_{4,t}, B_{5,t}, P_{1,t}, \dots$). All vectors have the same length which represents the number of GoPs in the encoded video trace. Depending on the trace's GoP pattern, the length of vector B changes. For instance if the trace uses the $G_{16}B_1$ pattern, then $B_t = [B_{1,t}, B_{2,t}, B_{3,t}, B_{4,t}, B_{5,t}, B_{6,t}, B_{7,t}, B_{8,t}]$, otherwise for $G_{16}B_7$ $B_t = [B_{1,t}, B_{2,t}, B_{3,t}, B_{4,t}, B_{5,t}, B_{6,t}, B_{7,t}, B_{8,t}, B_{9,t}, B_{10,t}, B_{11,t}, B_{12,t}, B_{13,t}, B_{14,t}]$.

As explained above, the initial part of our work follows the rationale that was used in [14]. We employ a linear regression-type model of prediction for each encoded movie based on the highest inter-frame correlation coefficients. An indicative example of the sets of equations for the prediction of the size of each B -frame is presented in Equation 3.3. The set is for the H.265 video trace "Harry Potter" in high quality and contains the frames with the highest correlation with the B -frame whose size needs to be predicted.

Harry Potter $G_{24}B_7$ QP=10

$$\begin{aligned}
\hat{B}_{1,t} &= \alpha_1 B_{15,t-1} + \gamma_1 B_{8,t-1} & \hat{B}_{11,t} &= \alpha_{11} B_{4,t} + \gamma_{11} P_{18,t-1} \\
\hat{B}_{2,t} &= \alpha_2 B_{16,t-1} + \gamma_2 B_{9,t-1} & \hat{B}_{12,t} &= \alpha_{12} B_{5,t} + \gamma_{12} B_{19,t-1} \\
\hat{B}_{3,t} &= \alpha_3 B_{17,t-1} + \gamma_3 B_{10,t-1} & \hat{B}_{13,t} &= \alpha_{13} B_{6,t} + \gamma_{13} B_{20,t-1} \\
\hat{B}_{4,t} &= \alpha_4 B_{18,t-1} + \gamma_4 P_{1,t} & \hat{B}_{14,t} &= \alpha_{14} B_{7,t} + \gamma_{14} B_{21,t-1} \\
\hat{B}_{5,t} &= \alpha_5 B_{19,t-1} + \gamma_5 B_{12,t-1} & \hat{B}_{15,t} &= \alpha_{15} B_{8,t} + \gamma_{15} B_{1,t} \\
\hat{B}_{6,t} &= \alpha_6 B_{20,t-1} + \gamma_6 B_{13,t-1} & \hat{B}_{16,t} &= \alpha_{16} B_{9,t} + \gamma_{16} B_{2,t} \\
\hat{B}_{7,t} &= \alpha_7 B_{21,t-1} + \gamma_7 B_{14,t-1} & \hat{B}_{17,t} &= \alpha_{17} B_{10,t} + \gamma_{17} B_{3,t} \\
\hat{B}_{8,t} &= \alpha_8 B_{1,t} + \gamma_8 B_{15,t-1} & \hat{B}_{18,t} &= \alpha_{18} B_{11,t} + \gamma_{18} B_{4,t} \\
\hat{B}_{9,t} &= \alpha_9 B_{2,t} + \gamma_9 B_{16,t-1} & \hat{B}_{19,t} &= \alpha_{19} B_{12,t} + \gamma_{19} B_{5,t} \\
\hat{B}_{10,t} &= \alpha_{10} B_{3,t} + \gamma_{10} B_{17,t-1} & \hat{B}_{20,t} &= \alpha_{20} B_{13,t} + \gamma_{20} B_{6,t} \\
& & \hat{B}_{21,t} &= \alpha_{21} B_{14,t} + \gamma_{21} B_{7,t}
\end{aligned} \tag{3.3}$$

We used the least squares method in Matlab to compute the coefficients α_j and γ_j with j ranging from 1 to the number of B -frames in a GoP (depending on the GoP pattern, this number can be equal to 8, 12, 14,15 for H.264 videos and 21 for the H.265 ones). The predicted sizes for all B -frames are derived by inserting the α_j and γ_j values into the corresponding equation set.

The intuitive approach is to derive an individual set of equations for each video trace, as each trace is different from the rest. However, the authors in [14] chose a different approach, for computational simplicity reasons: they found the MPEG-4 trace for which the prediction was most accurate and they used the set of equations of that trace for all traces in their study. Then, depending on the individual results, they “tweaked” their model by changing a couple of equations for specific B -frames, if the prediction accuracy was not satisfactory for a trace. In our previous work [15] we used both approaches, i.e., the one with a separate set of equations for each trace and the one where “one set of equations fits all” for H.264 movie traces. In this work we tested these 2 approaches for H.265 videos and compared the results.

In order to evaluate the accuracy of our predictions and to compute the difference from the actual value of the B -frame we used the Relative Percentage Error (RPE) defined in equation 3.4.

$$RPE = \frac{\sum_{m=1}^L \epsilon_m}{\sum_{m=1}^L X_m} \times 100\% \tag{3.4}$$

where L is the number of B -frames in the encoded video, X is the actual size of the frame and ϵ is the marginal error, defined as the difference between the predicted size of the frame and its actual size, as shown in equation 3.5.

$$\epsilon = \bar{X}_m - X_m \quad (3.5)$$

where \bar{X}_m is the predicted size of the B -frame and X_m is the actual size.

3.2.2 Results and Comparison

We present the RPE results for the H.264 traces computed in my Diploma thesis [15], both for the approach with an individual set of equations for each trace and for the “one set of equations fits all” approach. The results for the first approach are presented in the second column of Table 3.3 and we observe that the “NBC news” QP=10 trace exhibits the lowest RPE. Following the approach of [14] (“one set of equations fits all”), we used this trace’s equations set (for $G_{16}B_1$, $G_{16}B_3$, $G_{16}B_7$, $G_{16}B_{15}$, respectively) for all H.264 traces. The results are presented in the third column of Table 3.3. It is clear from the results that, with very few exceptions, the RPE is very high for both approaches.

Our new RPE results for H.265 traces for both approaches, are presented in 3.4. For the second approach, the set of equations used was the one corresponding to the “Speed” QP=10 trace, which exhibits the lowest RPE. As shown from the results, in seven out of the eight cases, the lowest quality versions of each movie (QP=40) are the ones with the larger RPE. This can be explained by the CoV values in Table 2.2, which are often consistently higher for QP=40 videos, denoting that the outliers clearly affect our results. Again, the RPE is very high for both approaches. The average RPE over all H.265 traces is 24.7% when an individual set of equations is used for each trace and 26.1% when “one set of equations fits all”.

It should also be noted that, as shown in both Tables 3.3 and 3.4, most of the RPE values are the same for both approaches. The reason is that the traces chosen for the “one set of equations fits all” approach (due to the fact that they have the lowest RPE) often have the same set of equations with the other video traces of the same GoP pattern.

Table 3.3
IFCC-based model - RPE Results for H.264 Video Traces

H.264	RPE (Model: One set for each trace)	RPE (Model: One set fits all)
Silence of the Lambs $G_{16}B_1$ -QP=10	73.33%	68.42%
Silence of the Lambs $G_{16}B_1$ -QP=28	85.54%	71.79%
Silence of the Lambs $G_{16}B_1$ -QP=48	15.35%	15.35%
Silence of the Lambs $G_{16}B_3$ -QP=10	37.41%	36.31%
Silence of the Lambs $G_{16}B_3$ -QP=28	47.17%	39.92%
Sil. of the Lambs $G_{16}B_3$ -QP=48	11.82%	11.82%
Sil. of the Lambs $G_{16}B_7$ -QP=10	16.70%	14.52%
Sil. of the Lambs $G_{16}B_7$ -QP=28	23.19%	23.19%
Sil. of the Lambs $G_{16}B_7$ -QP=48	10.95%	10.95%
Sil. of the Lambs $G_{16}B_{15}$ -QP=10	10.22%	10.22%
Sil. of the Lambs $G_{16}B_{15}$ -QP=28	14.45%	14.45%
Sil. of the Lambs $G_{16}B_{15}$ -QP=48	10.00%	10.00%
NBC news $G_{16}B_1$ -QP=10	5.47%	5.47%
NBC news $G_{16}B_1$ -QP=28	39.99%	39.12%
NBC news $G_{16}B_1$ -QP=48	28.90%	29.52%
NBC news $G_{16}B_3$ -QP=10	4.94%	4.94%
NBC news $G_{16}B_3$ -QP=28	28.06%	28.06%
NBC news $G_{16}B_3$ -QP=48	23.04%	23.04%
NBC news $G_{16}B_7$ -QP=10	4.53%	4.53%
NBC news $G_{16}B_7$ -QP=28	20.28%	20.28%
NBC news $G_{16}B_7$ -QP=48	19.53%	19.53%
NBC news $G_{16}B_{15}$ -QP=10	3.79%	3.79%
NBC news $G_{16}B_{15}$ -QP=28	14.76%	14.76%
NBC news $G_{16}B_{15}$ -QP=48	16.01%	16.01%
Star Wars IV $G_{16}B_1$ -QP=10	46.39%	42.78%
Star Wars IV $G_{16}B_1$ -QP=28	47.15%	47.30%
Star Wars IV $G_{16}B_1$ -QP=48	18.24%	18.22%
Star Wars IV $G_{16}B_3$ -QP=10	21.93%	21.93%
Star Wars IV $G_{16}B_3$ -QP=28	29.20%	29.20%
Star Wars IV $G_{16}B_3$ -QP=48	14.43%	14.32%
Star Wars IV $G_{16}B_7$ -QP=10	13.34%	13.34%
Star Wars IV $G_{16}B_7$ -QP=28	19.07%	19.07%
Star Wars IV $G_{16}B_7$ -QP=48	12.57%	12.57%
Star Wars IV $G_{16}B_{15}$ -QP=10	8.96%	8.96%
Star Wars IV $G_{16}B_{15}$ -QP=28	12.88%	12.88%
Star Wars IV $G_{16}B_{15}$ -QP=48	10.51%	10.51%

Table 3.4
IFCC-based model - RPE Results for H.265 Video Traces

H.265	RPE (Model: One set for each trace)	RPE (Model: One set fits all)
BigBuckBunny-QP=10	30.80%	29.51%
BigBuckBunny-QP=25	34.22%	46.45%
BigBuckBunny-QP=40	36.46%	60.87%
BluePlanet-QP=10	24.16%	25.05%
BluePlanet-QP=25	19.34%	19.34%
BluePlanet-QP=40	19.48%	19.48%
Speed-QP=10	9.67%	9.67%
Speed-QP=25	25.68%	25.68%
Speed-QP=40	29.47%	29.47%
LakeHouse-QP=10	17.39%	17.39%
LakeHouse-QP=25	23.13%	23.13%
LakeHouse-QP=40	27.06%	27.06%
FindingNeverland-QP=10	11.53%	11.53%
FindingNeverland-QP=25	25.04%	25.04%
FindingNeverland-QP=40	31.28%	31.28%
ElephantsDream-QP=10	17.54%	17.54%
ElephantsDream-QP=25	25.32%	24.77%
ElephantsDream-QP=40	34.05%	33.46%
TearsOfSteel-QP=10	11.79%	10.12%
TearsOfSteel-QP=25	23.76%	23.76%
TearsOfSteel-QP=40	41.39%	40.08%
HarryPotter-QP=10	12.98%	12.98%
HarryPotter-QP=25	29.90%	29.90%
HarryPotter-QP=40	32.31%	32.31%

Chapter 4

Simulated Annealing

4.1 Introduction

Simulated Annealing [38] is a probabilistic method that can be used as a metaheuristic in complex optimization problems. It has the advantage, in comparison to other techniques, that it can avoid getting caught at local maxima-solutions that are better than any others nearby, but aren't the very best. A good example has been given in [39] as it is presented in Figure 4.1 .

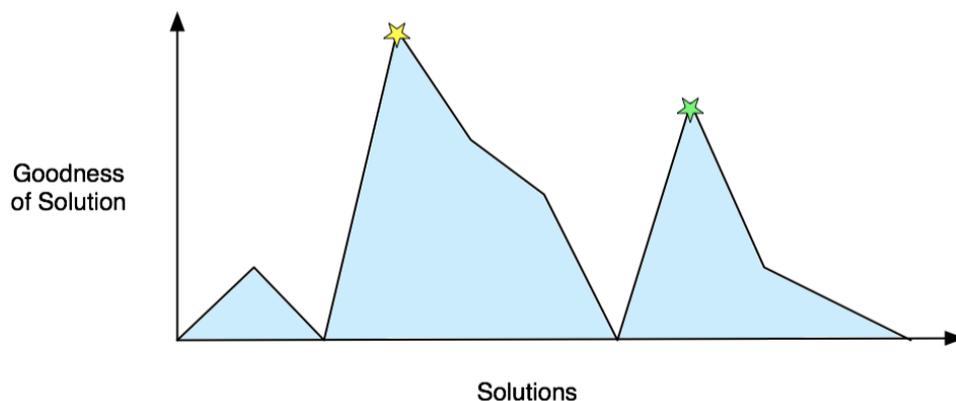


Fig. 4.1 Example of a local vs absolute maximum

A simple optimization algorithm searches for the best solution by generating a random initial solution and “exploring” the area nearby. In this example if the algorithm finds its way to the green star on the right, it won't move away from it because all of the neighboring solutions are worse, but the green star is a local maximum. On the other hand, Simulated annealing sometimes decides (randomly) to keep the worse

solution and not always avoid it. In that way it is possible to escape local maxima solutions and find its way to the Absolute maxima.

The basic Algorithm steps as presented in [39] is:

1. First, generate a random solution
2. Calculate its cost
3. Generate a random neighboring solution
4. Calculate the new solution's cost
5. Compare them:
 - If $c_{new} < c_{old}$: move to the new solution
 - If $c_{new} > c_{old}$: **maybe** move to the new solution
6. Repeat steps 3-5 above until an acceptable solution is found or you reach some maximum number of iterations.

4.2 Using Simulated Annealing for prediction

4.2.1 Implementation

In our work, we used Simulated Annealing to find the best equation set for the prediction of B-frames' sizes. In [14], the authors changed manually certain equations in specific traces where their RPE results were high. They chose the specific equations because of the characteristic of certain *B*-frames to have a higher autocorrelation with their subsequent *P*-frame. What we propose is the use of Simulated Annealing in order to test multiple equation sets and finally choose the best set of equations (leading to the lowest RPE) for the prediction of the size of *B*-frames in every video trace. This approach can be very helpful not only in the transmission of a specific video, but also in the case of transmitting other, similar in nature videos for which the content is not a priori known (for example, live webcasting); the same equation set can be used for similar videos in such a case.

In the first step of the algorithm, we assume that the most relevant frames for the prediction of a *B*-frame size are the two previous *B*-frames (as they often have the highest correlation), and we form the equation set accordingly. The second step is to calculate the cost, which in our case is the RPE. of the prediction using the above

solution. The third step is to generate a random neighboring solution. Thus, we choose randomly one of the equations and only one of the two predictors of this equation. The selected value is replaced randomly by another predictor (another *B*-frame or *P*-frame). We compute the new solution's cost in the fourth step. In the fifth step, we compare the old cost with the new one. In order to compare the costs and decide which solution to keep, the algorithm calculates the "acceptance probability" p , as shown in equation 4.1

$$p = \exp\left(-\frac{\Delta_c}{T}\right) \quad (4.1)$$

where Δ_c is the cost change, with the new solution, and T is the current "temperature" of the system [40]. The temperature is a function of the iteration number in the algorithm. Similarly to the physical annealing in solids, as the temperature of the system decreases, so will the probability of accepting a worse move decrease. The temperature starts at 1.0 and decreases by a constant k , in each iteration. We used $k = 0.9$, as in [38]. The acceptance probability p is then compared against a random number between 0 and 1, and if p is larger, the system moves to the new solution (which may be worse, and this is a temporary trade-off that we are willing to accept in order for the solution not to get trapped in a local optimum). The procedure is repeated until the maximum number of iterations (500, in our work) is reached, and a final set of equations is chosen. The implementation described above is presented shortly in Algorithm 1

4.2.2 Results and Complexity

Our results, presented in Tables 4.1, 4.2 and 4.3 for all types of encodings, show that the use of Simulated Annealing leads to a substantial decrease of RPE. On average, the improvement for MPEG-4 traces is 10.4% when compared against the "one set fits all" approach and 4% against the approach of manually changing specific equations that was used in [14] (the values missing from the respective column in 4.1 are ones for which [14] did not provide any results). For H.264 the improvement offered by Simulated Annealing is 4.7% when compared against the "one set fits all" approach and 6.3% against the "one set for each trace" approach. For H.265 the improvement offered by Simulated Annealing is 7.9% when compared against the "one set fits all" approach and 9.2% against the "one set for each trace" approach.

Algorithm 1 Steps of prediction using Simulated Annealing

```

//neighbor() changes randomly one of the predictors in one equations of the set
//cost() computes the RPE
//acceptance_probability() returns the result of equation 4.1
//oldSet denotes the equation set using the size of the two previous B-frames
1: Temp=1
2: Temp_min=0.00001
3: p=0.9
4: OldCost=cost(oldSet)
5: for Temp>Temp_min do counter=1;
6:   while counter<=500 do
7:     newSet=neighbor(oldSet)
8:     newCost=cost(newSet)
9:     ap=acceptance_probability(OldCost, newCost, T)
10:    if ap>random then
11:      oldSet=newSet
12:      oldCost=newCost
13:    end if
14:    counter++
15:    T=T*p
16:  end while
17: end for

```

Table 4.1
RPE Results for MPEG-4 Video Traces

MPEG-4	RPE (%)	RPE (%)	RPE (%)
	One set fits all	Manual change of specific equations	Simulated Annealing
<i>Silence of the Lambs HQ</i>	7.47	7.30	7.39
<i>Silence of the Lambs MQ</i>	12.46	-	12.05
<i>Silence of the Lambs LQ</i>	17.1	16.64	15.87
<i>Aladdin HQ</i>	22.4	20.72	18.28
<i>Aladdin MQ</i>	26.06	-	21.26
<i>Aladdin LQ</i>	28.7	-	23.97
<i>Ski HQ</i>	7.09	6.95	6.95
<i>Simpsons HQ</i>	10.78	9.53	9.06

Simulated Annealing has a complexity [41] of $O((n^2 + n)\log n)$, where n is the number of B -frames in a GoP, in our system (and the number of equations in an equation set, respectively). Due to the small number of B -frames in a GoP, the

complexity of the Simulated Annealing algorithm is small, and does not create any runtime problems in its execution.

Table 4.3
RPE RESULTS FOR H.265 VIDEO TRACES

H.265	RPE (Model: One set for each trace)	RPE (Model: One set for each trace)	RPE Simulated Annealing
BigBuckBunny QP=10	30.796%	29.5126%	17.34383%
BigBuckBunny QP=25	34.217%	46.446%	30.09475%
BigBuckBunny QP=40	36.454%	60.872%	29.33684%
BluePlanet QP=10	24.156%	25.0463%	18.59828%
BluePlanet QP=25	19.338%	19.338%	17.88015%
BluePlanet QP=40	19.483%	19.483%	18.27384%
Speed QP=10	9.668%	9.668%	6.301531%
Speed QP=25	25.681%	25.681%	16.75723%
Speed QP=40	29.469%	29.469%	17.76566%
LakeHouse QP=10	17.394%	17.394%	11.50432%
LakeHouse QP=25	23.128%	23.128%	15.14469%
LakeHouse QP=40	27.063%	27.063%	17.59302%
FindingNeverland QP=10	11.533%	11.533%	8.089894%
FindingNeverland QP=25	25.040%	25.040%	14.93698%
FindingNeverland QP=40	31.277%	31.277%	17.65051%
ElephantsDream QP=10	17.543%	17.543%	11.55108%
ElephantsDream QP=25	25.318%	24.770%	16.81154%
ElephantsDream QP=40	34.047%	33.462%	19.27894%
TearsOfSteel QP=10	11.786%	10.119%	6.638212%
TearsOfSteel QP=25	23.7580%	23.7580%	19.34766%
TearsOfSteel QP=40	41.394%	40.0757%	23.96085%
HarryPotter QP=10	13.283%	13.258%	9.957683%
HarryPotter QP=25	29.897%	29.897%	19.55038%
HarryPotter QP=40	32.310%	32.310%	20.45322%

Table 4.2
RPE Results for H.264 Video Traces

H.264	RPE (Model: One set for each trace)	RPE (Model: One set for each trace)	RPE Simulated Annealing
Silence of the Lambs G16B1-Qp=10	73.3295%	68.421%	64.98657%
Silence of the Lambs G16B1-Qp=28	85.537%	71.794%	60.47636%
Silence of the Lambs G16B1-Qp=48	15.349%	15.346%	14.62761%
Silence of the Lambs G16B3-Qp=10	37.4105%	36.3062%	27.7018%
Silence of the Lambs G16B3-Qp=28	47.168%	39.919%	34.18315%
Silence of the Lambs G16B3-Qp=48	11.8193%	11.8193%	11.57454%
Silence of the Lambs G16B7-Qp=10	16.698%	14.5200%	14.42429%
Silence of the Lambs G16B7-Qp=28	23.192%	23.192%	20.70621%
Silence of the Lambs G16B7-Qp=48	10.945%	10.945%	10.60747%
Silence of the Lambs G16B15-Qp=10	10.218%	10.218%	9.038225%
Silence of the Lambs G16B15-Qp=28	14.445%	14.445%	13.42066%
Silence of the Lambs G16B15-Qp=48	10.005%	10.005%	9.770668%
NBC news G16B1-Qp=10	5.474%	5.474%	5.463069%
NBC news G16B1-Qp=28	39.993%	39.119%	38.91292%
NBC news G16B1-Qp=48	28.902%	29.524%	27.63503%
NBC news G16B3-Qp=10	4.9427%	4.9427%	4.783985%
NBC news G16B3-Qp=28	28.0646%	28.0646%	27.44421%
NBC news G16B3-Qp=48	23.0409%	23.0409%	22.41409%
NBC news G16B7-Qp=10	4.53048%	4.53048%	4.474553%
NBC news G16B7-Qp=28	20.279%	20.279%	19.57994%
NBC news G16B7-Qp=48	19.534%	19.534%	18.957%
NBC news G16B15-Qp=10	3.7882%	3.7882%	3.871235%
NBC news G16B15-Qp=28	14.7639%	14.7639%	14.7338%
NBC news G16B15-Qp=48	16.0115%	16.0115%	15.78022%
Star Wars IV G16B1-Qp=10	46.3945%	42.783%	41.89814%
Star Wars IV G16B1-Qp=28	47.147%	47.297%	47.14944%
Star Wars IV G16B1-Qp=48	18.237%	18.218%	16.53172%
Star Wars IV G16B3-Qp=10	21.9308%	21.9308%	21.49285%
Star Wars IV G16B3-Qp=28	29.2029%	29.2029%	27.71664%
Star Wars IV G16B3-Qp=48	14.4279%	14.3129%	13.68016%
Star Wars IV G16B7-Qp=10	13.341%	13.341%	12.81866%
Star Wars IV G16B7-Qp=28	19.0677%	19.0677%	18.27941%
Star Wars IV G16B7-Qp=48	12.5749%	12.5749%	11.80185%
Star Wars IV G16B15-Qp=10	8.959%	8.959%	8.713207%
Star Wars IV G16B15-Qp=28	12.876%	12.876%	12.34638%
Star Wars IV G16B15-Qp=48	10.514%	10.514%	10.4214%

Chapter 5

Markovian Model

It is important to mention here that even with the use of Simulated Annealing there are quite a few cases where the achieved RPE was high. The difficulty of these approaches to predict with high accuracy the size of upcoming B-frames led us to try to understand the unsatisfactory results.

5.1 Outliers

While analyzing the datasets and specifically the points where the prediction was least accurate, we found that in many of the H.265 traces and in several H.264 traces there was a significant number of *B*-frames with much smaller size than the mean size of *B*-frames. These very small *B*-frames could be placed in between large B-frames which makes their prediction more difficult. They were also randomly placed in the video trace and due to their very small size they couldn't be well predicted using the set of equations. As a result, RPE was significantly increased when these small *B*-frames' sizes were inaccurately predicted.

Tables 5.1 and 5.2 show the extent of this problem for our predictions, for H.264 and H.265 traces, respectively. We have set a threshold, equal to 10% of the mean *B*-frame size, to separate small *B*-frames (those with size smaller than 10% of the mean) from the rest. In order to choose the threshold, we have experimented with a number of different percentages. If a very small value is chosen as a threshold, e.g., 1% of the mean *B*-frame size, the influence of small *B*-frames on the results cannot be adequately shown, as there are few traces in which so small *B*-frames are present. The bulk of the “problematic” small *B*-frames have sizes between 5% and 10% of the mean *B*-frame size, and this was the reason we chose 10% as a threshold. In Table 5.1 and Table 5.2 we present the RPE results for the prediction of only the “small”

B -frames both for threshold set to 1% and 10% of the mean size of B -frames (in some cases there were no “small” B -frames and this is indicated in the Tables).

Table 5.1
RPE for prediction of “small” B -frames’ sizes for H.264 videos

H.264	RPE Threshold= 1/10*Mean Size of B-frames	RPE Threshold= 1/100*Mean Size of B-frames
Silence of the Lambs $G_{16}B_1$ -Qp=10	4248.2%	17787%
Silence of the Lambs $G_{16}B_1$ -Qp=28	No small frames	No small frames
Silence of the Lambs $G_{16}B_1$ -Qp=48	No small frames	No small frames
Silence of the Lambs $G_{16}B3$ -Qp=10	4664.9%	19637%
Silence of the Lambs $G_{16}B3$ -Qp=28	294.6967%	No small frames
Silence of the Lambs $G_{16}B3$ -Qp=48	No small frames	No small frames
Silence of the Lambs $G_{16}B7$ -Qp=10	2498.4%	16250%
Silence of the Lambs $G_{16}B7$ -Qp=28	294.39%	No small frames
Silence of the Lambs $G_{16}B7$ -Qp=48	No small frames	No small frames
Silence of the Lambs $G_{16}B_{15}$ -Qp=10	1233.4%	14729%
Silence of the Lambs $G_{16}B_{15}$ -Qp=28	165.573%	No small frames
Silence of the Lambs $G_{16}B_{15}$ -Qp=48	No small frames	No small frames
NBC news $G_{16}B_1$ -Qp=10	No small frames	No small frames
NBC news $G_{16}B_1$ -Qp=28	393.6143%	No small frames
NBC news $G_{16}B_1$ -Qp=48	No small frames	No small frames
NBC news $G_{16}B3$ -Qp=10	No small frames	No small frames
NBC news $G_{16}B3$ -Qp=28	238.83%	No small frames
NBC news $G_{16}B3$ -Qp=48	No small frames	No small frames
NBC news $G_{16}B7$ -Qp=10	No small frames	No small frames
NBC news $G_{16}B7$ -Qp=28	182.27%	No small frames
NBC news $G_{16}B7$ -Qp=48	No small frames	No small frames
NBC news $G_{16}B_{15}$ -Qp=10	No small frames	No small frames
NBC news $G_{16}B_{15}$ -Qp=28	154.75%	No small frames
NBC news $G_{16}B_{15}$ -Qp=48	No small frames	No small frames
Star Wars IV $G_{16}B_1$ -Qp=10	1760.6%	9910.1%
Star Wars IV $G_{16}B_1$ -Qp=28	No small frames	No small frames
Star Wars IV $G_{16}B_1$ -Qp=48	No small frames	No small frames
Star Wars IV $G_{16}B3$ -Qp=10	1284%	9669.9%
Star Wars IV $G_{16}B3$ -Qp=28	130.45%	No small frames
Star Wars IV $G_{16}B3$ -Qp=48	No small frames	No small frames
Star Wars IV $G_{16}B7$ -Qp=10	627.75%	8651.2%
Star Wars IV $G_{16}B7$ -Qp=28	119.54%	No small frames
Star Wars IV $G_{16}B7$ -Qp=48	No small frames	No small frames
Star Wars IV $G_{16}B_{15}$ -Qp=10	321.6057%	6172.2%
Star Wars IV $G_{16}B_{15}$ -Qp=28	113.7%	No small frames
Star Wars IV $G_{16}B_{15}$ -Qp=48	No small frames	No small frames

It is clear from both Tables that the prediction of “small” B -frames’ sizes is completely inaccurate when using the IFCCs methodology, and is responsible for significantly increasing the average RPE value. The problem is more pronounced in

Table 5.2
RPE for prediction of “small” B-frames’ sizes for H.265 videos

H.265	RPE Threshold= 1/10*Mean Size of B-frames	RPE Threshold= 1/100*Mean Size of B-frames
BigBuckBunny QP=10	408.04%	5133%
BigBuckBunny QP=25	340.39%	1615.3
BigBuckBunny QP=40	188.65%	No small frames
BluePlanet QP=10	205.30%	8567.5%
BluePlanet QP=25	74.93%	No small frames
BluePlanet QP=40	No small frames	No small frames
Speed QP=10	664.42%	11528%
Speed QP=25	171.58%	2901.5%
Speed QP=40	196.47%	No small frames
LakeHouse QP=10	472.16%	11413%
LakeHouse QP=25	98.28%	No small frames
LakeHouse QP=40	86.06%	No small frames
FindingNeverland QP=10	833.10%	18188%
FindingNeverland QP=25	125.23%	990.14%
FindingNeverland QP=40	128.57%	No small frames
ElephantsDream QP=10	167.73%	6441.2%
ElephantsDream QP=25	195.51%	1782.3%
ElephantsDream QP=40	180.43%	No small frames
TearsOfSteel QP=10	1196.2%	14699%
TearsOfSteel QP=25	161.70%	3279.5%
TearsOfSteel QP=40	243.36%	No small frames
HarryPotter QP=10	533.12%	43624%
HarryPotter QP=25	179.59%	1991.8%
HarryPotter QP=40	132.41%	No small frames

the case of H.265 traces than in the case of H.264 traces, as there are several H.264 traces that do not contain very small *B*-frames.

We followed the same procedure to find out if there were also “big” B-frames that could affect our RPE result. In order to have correspondence between the “small” and the “big” *B*-frames we tested if there were B-frames sizes bigger than 10 times the mean *B*-frame size. It appears that in the majority of the movie traces there were only few such big B-frames or none at all. However, by analyzing the data sets and in particular the exact frames which caused bad prediction results, we found that in the cases where such “big” frames exist, they clearly affect the overall RPE result.

5.2 Big and small B-Frames

The observation of the problem, caused by B -frames' sizes which were far from the mean B -frame' size, led us to split the B -frames of a GoP in two subsets, one containing "Small B -frames" and the other containing "Big B -frames". We first marked as "Small B -frames" (SBFs) all B -frames with smaller size than the median value for each trace and as "Big B -frames" (BBFs), all B -frames with bigger size than the median value, for each trace. If the majority of the frames of a specific B -frame (e.g., B_1) were marked as BBFs then all B_1 frames were marked as BBFs. Likewise, if the majority of frames of a specific B -frame (e.g., B_2) were marked as SBFs then all B_2 frames were marked as SBFs. After separating the B -frames of the GoP in BBFs and SBFs in the aforementioned manner, we use the prediction equations to predict the size of a B -frame. If the B -frame has been marked as BBF we can only use in the equations the sizes of previous BBFs or P-frames (similarly for SBFs).

We have computed the autocorrelation between SBFs and the autocorrelation between BBFs with equation 3.2. Then we computed the correlation coefficient between SBFs and P-frames and between BBFs and P-frames using equations 3.1. Finally we created each equation in the way described in Section 3.2. An example of the set of equations, used for prediction is presented in 5.1.

$$\begin{array}{l}
 \text{Harry Potter } G_{24}B_7 \text{ QP=10} \\
 \hat{B}_{1,t} = \alpha_1 B_{21,t-1} + \gamma_1 B_{17,t-1} \\
 \hat{B}_{2,t} = \alpha_5 B_{16,t-1} + \gamma_5 P_{1,t} \\
 \hat{B}_{3,t} = \alpha_2 B_{1,t-1} + \gamma_2 B_{19,t-1} \\
 \hat{B}_{4,t} = \alpha_5 B_{18,t-1} + \gamma_5 P_{1,t} \\
 \hat{B}_{5,t} = \alpha_3 B_{3,t-1} + \gamma_3 B_{21,t-1} \\
 \hat{B}_{6,t} = \alpha_5 B_{20,t-1} + \gamma_5 P_{1,t} \\
 \hat{B}_{7,t} = \alpha_4 B_{5,t} + \gamma_4 B_{1,t} \\
 \hat{B}_{8,t} = \alpha_5 B_{7,t} + \gamma_5 B_{3,t} \\
 \hat{B}_{9,t} = \alpha_5 B_{2,t} + \gamma_5 P_{2,t} \\
 \hat{B}_{10,t} = \alpha_5 B_{8,t} + \gamma_5 B_{5,t} \\
 \hat{B}_{11,t} = \alpha_5 P_{1,t} + \gamma_5 P_{2,t} \\
 \hat{B}_{12,t} = \alpha_5 B_{10,t} + \gamma_5 B_{7,t} \\
 \hat{B}_{13,t} = \alpha_5 B_{6,t} + \gamma_5 P_{2,t} \\
 \hat{B}_{14,t} = \alpha_5 B_{12,t} + \gamma_5 B_{8,t} \\
 \hat{B}_{15,t} = \alpha_5 B_{14,t} + \gamma_5 B_{10,t} \\
 \hat{B}_{16,t} = \alpha_5 B_{9,t} + \gamma_5 P_{2,t} \\
 \hat{B}_{17,t} = \alpha_5 B_{15,t} + \gamma_5 B_{12,t} \\
 \hat{B}_{18,t} = \alpha_5 B_{11,t} + \gamma_5 P_{2,t} \\
 \hat{B}_{19,t} = \alpha_5 B_{17,t} + \gamma_5 B_{14,t} \\
 \hat{B}_{20,t} = \alpha_5 B_{13,t} + \gamma_5 P_{2,t} \\
 \hat{B}_{21,t} = \alpha_5 B_{19,t} + \gamma_5 B_{15,t}
 \end{array} \tag{5.1}$$

where the $B_1, B_3, B_5, B_7, B_8, B_{10}, B_{12}, B_{14}, B_{15}, B_{17}, B_{19}, B_{21}$ are marked as SBFs, thus their prediction equations can use the sizes of B -frames that belong to this category. The rest of the B -frames are marked as BBFs and follow the same rule. We used the example of “Harry Potter” with QP=10 in order to make to compare the set with equations 3.3.

We computed the RPE results for this approach using equation 3.4, which was shown to yield a significant improvement for H.265 traces (4% smaller RPE, on average compared with the approach “one set for each trace”). However, it did not provide an improvement for H.264 traces, on average (the improvements for some traces were counterbalanced by the deterioration of the results for others); the reason was the above-mentioned fact that several H.264 traces do not contain very small B -frames. Tables 5.3 and 5.4 present the results of this model compared with our previous results for H.265 and H.264 compression standards respectively.

Table 5.3
RPE results for H.265 video traces using BBFs and SBFs

H.265	RPE One set for each trace	RPE Simulated Annealing	RPE BBFs and SBFs
BigBuckBunny QP=10	30.796%	17.34383%	11.0043%
BigBuckBunny QP=25	34.217%	30.09475%	25.2815%
BigBuckBunny QP=40	36.454%	29.33684%	29.9106%
BluePlanet QP=10	24.156%	18.59828%	18.2724%
BluePlanet QP=25	19.338%	17.88015%	31.1500%
BluePlanet QP=40	19.483%	18.27384%	34.7280%
Speed QP=10	9.668%	6.301531%	22.6038%
Speed QP=25	25.681%	16.75723%	20.5461%
Speed QP=40	29.469%	17.76566%	19.5289%
LakeHouse QP=10	17.394%	11.50432%	7.6100%
LakeHouse QP=25	23.128%	15.14469%	21.6718%
LakeHouse QP=40	27.063%	17.59302%	19.0561%
FindingNeverland QP=10	11.533%	8.089894%	15.6904%
FindingNeverland QP=25	25.040%	14.93698%	22.5612%
FindingNeverland QP=40	31.277%	17.65051%	16.2215 %
ElephantsDream QP=10	17.543%	11.55108%	10.7140%
ElephantsDream QP=25	25.318%	16.81154%	24.0476%
ElephantsDream QP=40	34.047%	19.27894%	29.77157%
TearsOfSteel QP=10	11.786%	6.638212%	11.8562%
TearsOfSteel QP=25	23.7580%	19.34766%	21.4559%
TearsOfSteel QP=40	41.394%	23.96085%	29.3622%
HarryPotter QP=10	13.283%	9.957683%	7.6400%
HarryPotter QP=25	29.897%	19.55038%	22.022%
HarryPotter QP=40	32.310%	20.45322%	26.700%

Table 5.4
RPE results for H.264 video traces using BBFs and SBFs

H.264	RPE One set for each trace	RPE Simulated Annealing	RPE BBFs and SBFs
Silence of the Lambs $G_{16}B_1$ -Qp=10	73.3295%	64.98657%	58.1905%
Silence of the Lambs $G_{16}B_1$ -Qp=10	73.3295%	64.98657%	58.1905%
Silence of the Lambs $G_{16}B_1$ -Qp=28	85.537%	60.47636%	64.8046%
Silence of the Lambs $G_{16}B_1$ -Qp=48	15.349%	14.62761%	15.7674%
Silence of the Lambs G16B3-Qp=10	37.4105%	27.7018%	31.4149%
Silence of the Lambs G16B3-Qp=28	47.168%	34.18315%	46.7803%
Silence of the Lambs G16B3-Qp=48	11.8193%	11.57454%	No small frames
Silence of the Lambs G16B7-Qp=10	16.698%	14.42429%	15.3638%
Silence of the Lambs G16B7-Qp=28	23.192%	20.70621%	24.1640%
Silence of the Lambs G16B7-Qp=48	10.945%	10.60747%	11.9249%
Silence of the Lambs $G_{16}B_{15}$ -Qp=10	10.218%	9.038225%	11.4160%
Silence of the Lambs $G_{16}B_{15}$ -Qp=28	14.445%	13.42066%	16.4545%
Silence of the Lambs $G_{16}B_{15}$ -Qp=48	10.005%	9.770668%	No small frames
NBC news $G_{16}B_1$ -Qp=10	5.474%	5.463069%	5.6766%
NBC news $G_{16}B_1$ -Qp=28	39.993%	38.91292%	41.3231%
NBC news $G_{16}B_1$ -Qp=48	28.902%	27.63503%	31.0690%
NBC news G16B3-Qp=10	4.9427%	4.783985%	5.6682%
NBC news G16B3-Qp=28	28.0646%	27.44421%	33.6449%
NBC news G16B3-Qp=48	23.0409%	22.41409%	29.4143%
NBC news G16B7-Qp=10	4.53048%	4.474553%	5.3572%
NBC news G16B7-Qp=28	20.279%	19.57994%	24.3551%
NBC news G16B7-Qp=48	19.534%	18.957%	21.4827%
NBC news $G_{16}B_{15}$ -Qp=10	3.7882%	3.871235%	4.4775%
NBC news $G_{16}B_{15}$ -Qp=28	14.7639%	14.7338%	17.8671%
NBC news $G_{16}B_{15}$ -Qp=48	16.0115%	15.78022%	19.8365%
Star Wars IV $G_{16}B_1$ -Qp=10	46.3945%	41.89814%	45.5847%
Star Wars IV $G_{16}B_1$ -Qp=28	47.147%	47.14944%	49.8962%
Star Wars IV $G_{16}B_1$ -Qp=48	18.237%	16.53172%	No small frames
Star Wars IV G16B3-Qp=10	21.9308%	21.49285%	27.1076%
Star Wars IV G16B3-Qp=28	29.2029%	27.71664%	37.7685%
Star Wars IV G16B3-Qp=48	14.4279%	13.68016%	No small frames
Star Wars IV G16B7-Qp=10	13.341%	12.81866%	16.1991%
Star Wars IV G16B7-Qp=28	19.0677%	18.27941%	23.2119%
Star Wars IV G16B7-Qp=48	12.5749%	11.80185%	No small frames
Star Wars IV $G_{16}B_{15}$ -Qp=10	8.959%	8.713207%	11.0438%
Star Wars IV $G_{16}B_{15}$ -Qp=28	12.876%	12.34638%	16.4520%
Star Wars IV $G_{16}B_{15}$ -Qp=48	10.514%	10.4214%	13.2407%

5.3 The Proposed Model

The results presented and discussed in Section 3.2.2 show that the linear regression modeling approach based on inter-frame correlation coefficients does not provide high accuracy for H.264 and H.265 traces, as it did in [14] for MPEG-4 traces. Even with the improvement we proposed and implemented in Section 5.2, the model is far from accurate. On the other hand, the splitting of B -frames into two subsets, one containing small and the other containing large sized frames, seemed to be a promising approach for handling the outliers (very small B -frames or very big B -frames) for H.265 traces and for those H.264 traces that contain a significant number of those B -frames.

Hence, we decided to keep this splitting of the B -frames in two subsets, but this time we performed the splitting for each type of B -frame in a GoP. Hence, each B -frame is split into a small and a large subset. Then, instead of using linear regression and exploiting IFCCs, we studied whether the B -frame sizes of each subset could be modeled with any of the well-known, from the literature, distributions for workload characterization and modeling.

- We tried to model the B -frames sizes using a Uniform density function

$$Uniform(x; \alpha, \beta) = \frac{1}{\beta - \alpha} \quad (5.2)$$

where α is the lower endpoint and β is the upper endpoint of our data.

- With exponential density function

$$Exponential(x; \mu) = \frac{1}{\mu} e^{-\frac{x}{\mu}} \quad (5.3)$$

where μ is the mean of the data

- With Gamma density function

$$Gamma(x; \alpha, \beta) = \frac{1}{\beta^\alpha \Gamma(\alpha)} x^{\alpha-1} e^{-\frac{x}{\beta}} \quad (5.4)$$

where α and β are the shape and scale parameters respectively. If m is the estimated mean and v the variance of the data set the parameters α and β can be estimated as $\beta = \frac{v}{m}$ and $\alpha = \frac{m}{\beta}$

- With Weibull density function

$$Weibull(x; \alpha, \beta) = \frac{\beta}{\alpha} \left(\frac{x}{\alpha} \right)^{\beta-1} e^{-\left(\frac{x}{\alpha}\right)^\beta} \quad (5.5)$$

where α is the scale parameter and β is the shape parameter computed using the maximum likelihood estimation method with the estimated parameters at the 95% significance level.

- With Lognormal density function

$$Lognormal(x; \mu, \sigma) = \frac{1}{x\sigma\sqrt{2\pi}} e^{-\frac{(\ln x - \mu)^2}{2\sigma^2}} \quad (5.6)$$

where μ and σ are the location and scale parameters. If m is the mean and v the variance then

$$\mu = \exp\left(\frac{m^2}{\sqrt{v + m^2}}\right) \quad (5.7)$$

and

$$\sigma = \sqrt{\log \frac{v}{m^2 + 1}} \quad (5.8)$$

The values of m and v were computed using the maximum likelihood estimation method with the estimated parameters at the 95% significance level.

- With Log-logistic density function

$$Log - logistic(x; \alpha, \beta) = \frac{\left(\frac{\beta}{\alpha}\right)\left(\frac{x}{\alpha}\right)^{\beta-1}}{\left(1 + \left(\frac{x}{\alpha}\right)^\beta\right)^2} \quad (5.9)$$

where α is the scale parameter and β is the shape parameter.

- Finally we tested the Pearson type V density function

$$PearsonV(x; \alpha, \beta) = \frac{1}{\beta\Gamma(\alpha)} \frac{e^{-\frac{\beta}{x}}}{\left(\frac{x}{\beta}\right)^{\alpha+1}} \quad (5.10)$$

where $\Gamma()$ is a Gamma function and α and β where computed by solving the system:

$$mu = \frac{\beta}{\alpha + 1} \quad (5.11)$$

$$v = \frac{\beta^2}{(\alpha - 1)^2(\alpha - 2)} \quad (5.12)$$

where μ and σ are the mean and variance of our data.

For every subset of B -frames we generated random values using all of the above mentioned distributions. After computing and sorting the new results, we compared them with the real values of sizes of B -frames. We used three different methods to evaluate the results. The first was the computation of the Relative Percentage Error (RPE), also used in the previous sections, given by equation 3.4. The second was the computation of the Mean Absolute Percentage Error (MAPE), which is defined in 5.13.

$$MAPE = \frac{1}{n} \sum_{t=1}^n \left| \frac{A_t - F_t}{A_t} \right| \times 100\% \quad (5.13)$$

where A_t is the actual value and F_t is the forecast value.

The third was the use of Q-Q plots. A Q-Q plot is a powerful goodness-of-fit test [42] which graphically compares two datasets in order to determine whether the datasets come from populations with a common distribution (if they do, the points of the plot should fall approximately along a 45 degree reference line). More specifically, a Q-Q plot is a plot of the quantiles of the data versus the quantiles of the fitted distribution. A z -quantile of X is any value x such that $P(X \leq x) = z$. We have plotted the quantiles of the real data with the respective quantiles of the various distribution fits.

The three methods do not agree in all cases on which distribution is the best fit for the population of B -frames' sizes in the small or big subset. In the cases of disagreement between the methods, we chose the distribution which was the best fit according to at least two of the three methods. In order to find the most accurate result: for each frame of the GoP we repeat the fitting procedure 50 times (enough times to compute an accurate mean but not too many times to make the program computational heavy). In the end we computed the mean of RPE and MAPE for each distribution and compare them. In 5.5 we present an example of the best fitting for both BBFs' and SBFs' distributions after 50 repetitions for "NBC news" video trace in $G_{16}B_1$, MQ(QP=28).

Even after 50 repetitions the results are not always the same. This happens due to the fact that the results of RPE are low for more than one distributions. This observation is very important because it gives us the degree of freedom to use only one distribution for the prediction of Big frames and only one for the prediction of Small frames, in each video trace. In this way the complexity decreases. We found for each individual trace, the one distribution which was the best or (marginally) second best fit for all small B -frames. The same was the case for all large B -frames.

Table 5.5
Best Fitting distributions for big and small B-frames After 50 repetitions for NBC
News $G_{16}B_1$ QP=28

Frames	Best Fit for Big B-frames (RPE results)	Best Fit for Big B-frames (MAPE results)	Best Fit for Small B-frames (RPE results)	Best Fit for Small B-frames (MAPE results)
B1	lognormal	loglogistic	uniform	gamma
B2	lognormal	loglogistic	uniform	gamma
B3	lognormal	loglogistic	uniform	gamma
B4	lognormal	loglogistic	uniform	gamma
B5	lognormal	loglogistic	uniform	gamma
B6	lognormal	loglogistic	uniform	gamma
B7	lognormal	loglogistic	uniform	gamma

For H.264 traces we found that the best distribution fits for large B -frames were provided by the lognormal, loglogistic or Pearson V distributions. The best distribution fits for small B -frames were provided by the weibull, Pearson V or uniform distributions. For H.265 traces we found that the best distribution fits for large B -frames were provided by the Pearson V or loglogistic distributions. The best distribution fits for small B -frames were provided by the weibull distribution for almost all traces, with only three exceptions where the uniform distribution provided the best fit. Table 5.6 and 5.7 present the Best Fitting Distributions for all the H.264 and H.265 video traces respectively. In two cases, “Silence f he Lambs” $G_{16}B_1$ QP=48 and “NBC news” $G_{16}B_1$ QP=48, the subset of “small” B -frames, calculated using median, contained multiple B -frames of the exact same value. Hence, the computed variation that is needed for the calculations of the distributions’ parameters, was equal to 0. In these cases there are no “small” or “big” frames and thus, we computed a general best fitting distribution for every B -frame in a GoP. For “Silence f he Lambs” $G_{16}B_1$ QP=48 we found that the distribution which best fits all the B -frame in a GoP is the Log-logistic. Likewise, for the trace “NBC news” $G_{16}B_1$ QP=48 the best fitting distribution is the Pearson type V.

After finding the best distribution fit for the small and big B -frames in each trace, we created a Markov chain model. We implemented a separate 2-state model for each B -frame. For example, for a movie with GoP structure $G_{16}B_1$ that has 8 B -frames in one GoP, we designed 8 different 2-state Markov chains, for a movie with GoP structure $G_{24}B_7$ that has 21 B -frames in one GoP we designed 21 different 2-state Markov chains etc. One state represents the BBFs and the other the SBFs. The transition from one state to the other is taking place using the 4 transition probabilities between small and big B -frames (BBF to SBF, SBF to BBF, BBF to BBF and SBF to SBF) which

Table 5.6
Best Fitting distributions for big and small B-frames for H.264 Video traces

H.264	Big Frames Best Fit (for every B-frame in the GoP)	Small Frames Best Fit (for every B-frame in the GoP)
Silence of the Lambs G16B1-Qp=10	Lognormal	Weibull
Silence of the Lambs G16B1-Qp=28	LogLogistic	Pearson type V
Silence of the Lambs G16B3-Qp=10	Pearson Type V	Weibull
Silence of the Lambs G16B3-Qp=28	LogLogistic	Weibull
Silence of the Lambs G16B3-Qp=48	LogLogistic	Lognormal
Silence of the Lambs G16B7-Qp=10	Pearson Type V	Weibull
Silence of the Lambs G16B7-Qp=28	LogLogistic	Weibull
Silence of the Lambs G16B7-Qp=48	Pearson Type V	Weibull
Silence of the Lambs G16B15-Qp=10	Pearson Type V	Weibull
Silence of the Lambs G16B15-Qp=28	Lognormal	Weibull
Silence of the Lambs G16B15-Qp=48	Pearson Type V	Weibull
NBC news G16B1-Qp=10	Pearson Type V	Weibull
NBC news G16B1-Qp=28	Lognormal	Uniform
NBC news G16B3-Qp=10	Pearson Type V	Weibull
NBC news G16B3-Qp=28	Lognormal	Uniform
NBC news G16B3-Qp=48	Pearson Type V	Pearson Type V
NBC news G16B7-Qp=10	Pearson Type V	Weibull
NBC news G16B7-Qp=28	Lognormal	Uniform
NBC news G16B7-Qp=48	Pearson Type V	Pearson Type V
NBC news G16B15-Qp=10	Pearson Type V	Weibull
NBC news G16B15-Qp=28	Lognormal	Uniform
NBC news G16B15-Qp=48	Pearson Type V	Pearson Type V
Star Wars IV G16B1-Qp=10	Pearson Type V	Weibull
Star Wars IV G16B1-Qp=28	Lognormal	Pearson Type V
Star Wars IV G16B1-Qp=48	LogLogistic	Pearson Type V
Star Wars IV G16B3-Qp=10	Pearson Type V	Weibull
Star Wars IV G16B3-Qp=28	LogLogistic	Weibull
Star Wars IV G16B3-Qp=48	Pearson Type V	Weibull
Star Wars IV G16B7-Qp=10	Pearson Type V	Uniform
Star Wars IV G16B7-Qp=28	Lognormal	Weibull
Star Wars IV G16B7-Qp=48	Pearson Type V	Weibull
Star Wars IV G16B15-Qp=10	Pearson Type V	Uniform
Star Wars IV G16B15-Qp=28	Lognormal	Weibull
Star Wars IV G16B15-Qp=48	Pearson Type V	Weibull

is calculated in the first part of our algorithm and most specifically when we split the B -frames into two subsets.

Hence, for each markov chain of every video trace, we generate an random number in the real interval $[0,1]$.After comparing it with the transition probabilities we predict, via the Markov chain model, the type of the B -frame will follow (small or big).The

Table 5.7
Best Fitting distributions for big and small B-frames for H.265 Video traces

H.265	Big Frames Best Fit (for every B-frame in the GoP)	Small Frames Best Fit (for every B-frame in the GoP)
BigBuckBunny QP=10	LogNormal	Weibull
BigBuckBunny QP=25	Pearson Type V	Weibull
BigBuckBunny QP=40	LogLogistic	Weibull
BluePlanet QP=10	Pearson Type V	Uniform
BluePlanet QP=25	LogNormal	Weibull
BluePlanet QP=40	Pearson Type V	Weibull
Speed QP=10	Pearson Type V	Weibull
Speed QP=25	Pearson Type V	Weibull
Speed QP=40	Pearson Type V	Weibull
LakeHouse QP=10	Pearson Type V	Weibull
LakeHouse QP=25	Pearson Type V	Weibull
LakeHouse QP=40	Pearson Type V	Weibull
FindingNeverland QP=10	Pearson Type V	Weibull
FindingNeverland QP=25	Pearson Type V	Weibull
FindingNeverland QP=40	Pearson Type V	Weibull
ElephantsDream QP=10	Pearson Type V	Uniform
ElephantsDream QP=25	LogLogistic	Weibull
ElephantsDream QP=40	LogLogistic	Weibull
TearsOfSteel QP=10	Pearson Type V	Uniform
TearsOfSteel QP=25	LogNormal	Weibull
TearsOfSteel QP=40	Pearson Type V	Weibull
HarryPotter QP=10	Pearson Type V	Weibull
HarryPotter QP=25	Pearson Type V	Weibull
HarryPotter QP=40	Pearson Type V	Weibull

respective (small or big) frame size is generated based on the best fit distribution for that trace.

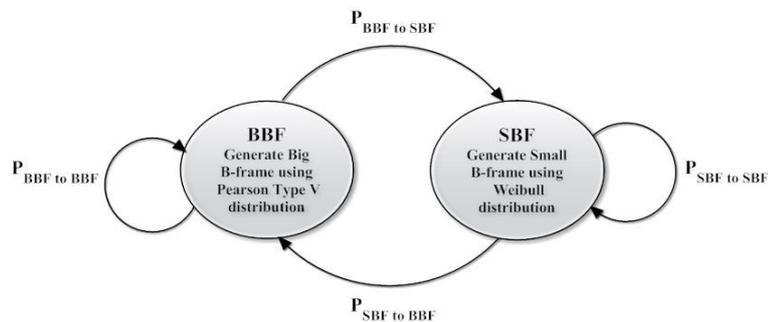


Fig. 5.1 Markov chain for prediction of the B1-frames for “Lake House” video trace in QP=10

In Figure 5.1 we present an example of the Markov chain used for the prediction of the B1-frames for the video trace “Lake House” in QP=10. For the specific trace, the big B -frames’ sizes are generated based on the Pearson V distribution, which was found to be the best fit for big B -frames, whereas the small B -frame sizes are generated based on the Weibull distribution, which was found to be the best fit for small B -frames.

The steps for our Markovian model are presented in Algorithm 2.

Algorithm 2 Steps of our proposed Markovian model using various distributions

```

1: procedure TESTING VARIOUS DISTRIBUTIONS
   //n denotes the B-frame number in a GoP
   // $P_{ab}, P_{bb}, P_{as}, P_{bs}$  :Pearson V parameters for big and small B-frames
   // $W_{ab}, W_{bb}, W_{as}, W_{bs}$  :Weibull parameters for big and small B-frames
2:   for  $i=1:1:n$  do
3:      $med = \text{median}(\text{B-frames})$ 
4:      $SBF \leftarrow \text{Bframes} < med$ 
5:      $BBF \leftarrow \text{Bframes} \geq med$ 
6:      $Pbs(n) = \text{transitionProb.fromBigtoSmall}$ 
7:      $Pbb(n) = \text{transitionProb.fromBigtoBig}$ 
8:      $Psb(n) = \text{transitionProb.fromSmalltoBig}$ 
9:      $Pss(n) = \text{transitionProb.fromSmalltoSmall}$ 
   //Generate random numbers with Pearson V distribution
10:     $PearsonB = \text{PearsonV}(P_{ab}, P_{bb})$ 
11:     $PearsonS = \text{PearsonV}(P_{as}, P_{bs})$ 
12:     $CriterionP1(i) = RPE(PearsonB, BBF)$ 
13:     $CriterionP2(i) = MAPE(PearsonB, BBF)$ 
14:     $CriterionP3(i) = QQplot(PearsonB, BBF)$ 
15:     $CriterionP1S(i) = RPE(PearsonS, SBF)$ 
16:     $CriterionP2S(i) = MAPE(PearsonS, SBF)$ 
17:     $CriterionP3S(i) = QQplot(PearsonS, SBF)$ 
   //Generate random numbers with Weibull distribution
18:     $WeibB = \text{Weibull}(W_{ab}, W_{bb})$ 
19:     $WeibS = \text{Weibull}(W_{as}, W_{bs})$ 
20:     $CriterionW1(i) = RPE(WeibB, BBF)$ 
21:     $CriterionW2(i) = MAPE(WeibB, BBF)$ 
22:     $CriterionW3(i) = QQplot(WeibB, BBF)$ 
23:     $CriterionW1S(i) = RPE(WeibS, SBF)$ 
24:     $CriterionW2S(i) = MAPE(WeibS, SBF)$ 
25:     $CriterionW3S(i) = QQplot(WeibS, SBF)$ 
   //..Generate random numbers using other distributions
26:   end for
27: end procedure

```

```

1: procedure COMPARE DISTRIBUTIONS AND CHOOSE THE BEST ONE
   //n denotes the B-frame number in a GoP
2:   for i=1:1:n do
3:      $BestDistBig(i) \leftarrow best(CriterionP1, CriterionP3, CriterionW1, ..)$ 
4:      $BestDistSmall(i) \leftarrow best(CriterionP1S, CriterionP2S, ...)$ 
5:   end for
6: end procedure

```

```

1: procedure PREDICTION AND MARKOV CHAIN
2:
3:   //n denotes the B-frame number in a GoP
4:    $state = 1$  ▷ 1 denotes the state for Big Frames
5:   for i=1:1:n do
6:      $count = 0$ 
7:     for all B(i) frames do ▷
8:        $r = rand$ 
9:       if state=1 and  $r > Pbs$  then
10:         $count = count + 1$ 
11:         $PredFrames(count) = BestDistBig(i)$ 
12:         $state = 1$ 
13:      else if state = 1 and  $r \leq Pbs$  then
14:         $count = count + 1$ 
15:         $PredFrames(count) = BestDistSmall(i)$ 
16:         $state = 0$  ▷ 1 denotes the state for Small Frames
17:      else if state = 0 and  $r \leq Pss$  then
18:         $count = count + 1$ 
19:         $PredFrames(count) = BestDistSmall(i)$ 
20:         $state = 0$ 
21:      else if state = 0 and  $r > Pss$  then
22:         $count = count + 1$ 
23:         $PredFrames(count) = BestDistBig(i)$ 
24:         $state = 1$ 
25:      end if
26:    end for
27:  end for
28: end procedure

```

5.4 Results

Tables 5.8 and 5.9 present our Markovian model's results in comparison to the IFCC-based method from [14], when a separate equation set is used for each trace. Our model is shown to clearly outperform the IFCC-based model. On average, the RPE

when using our model is 13.8% smaller for H.264 traces (an average improvement of 60.2%) and 14.82% smaller for H.265 traces (an improvement of 59.87%).

Table 5.8
Markov Model- RPE results for H.265 Video Traces

H.265	RPE (Model: One set for each trace)	RPE Markovian Model
BigBuckBunny QP=10	30.796%	11.3591%
BigBuckBunny QP=25	34.217%	11.9875%
BigBuckBunny QP=40	36.454%	14.1404%
BluePlanet QP=10	24.156%	4.3013%
BluePlanet QP=25	19.338%	14.0182%
BluePlanet QP=40	19.483%	16.9615%
Speed QP=10	9.668%	2.5218%
Speed QP=25	25.681%	4.4298%
Speed QP=40	29.469%	4.7787%
LakeHouse QP=10	17.394%	3.9638%
LakeHouse QP=25	23.128%	5.7637%
LakeHouse QP=40	27.063%	6.6876%
FindingNeverland QP=10	11.533%	2.7957%
FindingNeverland QP=25	25.040%	6.1285%
FindingNeverland QP=40	31.277%	6.6824%
ElephantsDream QP=10	17.543%	17.1325%
ElephantsDream QP=25	25.318%	23.2173%
ElephantsDream QP=40	34.047%	16.3487%
TearsOfSteel QP=10	11.786%	8.7376%
TearsOfSteel QP=25	23.7580%	24.5541%
TearsOfSteel QP=40	41.394%	13.8142%
HarryPotter QP=10	13.283%	2.5480%
HarryPotter QP=25	29.897%	6.7253%
HarryPotter QP=40	32.310%	8.7806%

As accurate as the modeling approach used might be, it is not of major importance if it cannot be used for prediction of future traffic. To ensure that overfitting does not take place with the use of our model we used “Repeated random sub-sampling validation”. Hence, we divided, multiple times, each of our trace datasets into two random subsets, one for training and one for testing. Each subset contains a number of GoPs of the video trace to train and evaluate our model. During the computation of the best fitting Distribution over the training GoPs the parameters of each distribution along with the transaction probabilities are saved in order to use them in the prediction

Table 5.9
Markov Model- RPE results for H.264 Video Traces

H.264	RPE (Model: One set for each trace)	RPE Markovian Model
Silence of the Lambs G16B1-Qp=10	73.3295%	10.6287%
Silence of the Lambs G16B1-Qp=28	85.537%	28.6345%
Silence of the Lambs G16B1-Qp=48	15.349%	
Silence of the Lambs G16B3-Qp=10	37.4105%	8.8341%
Silence of the Lambs G16B3-Qp=28	47.168%	19.4561%
Silence of the Lambs G16B3-Qp=48	11.8193%	11.6337%
Silence of the Lambs G16B7-Qp=10	16.698%	7.0562%
Silence of the Lambs G16B7-Qp=28	23.192%	15.7460%
Silence of the Lambs G16B7-Qp=48	10.945%	11.1072%
Silence of the Lambs G16B15-Qp=10	10.218%	5.0496%
Silence of the Lambs G16B15-Qp=28	14.445%	14.7654%
Silence of the Lambs G16B15-Qp=48	10.005%	7.0128%
NBC news G16B1-Qp=10	5.474%	1.9479%
NBC news G16B1-Qp=28	39.993%	11.3712%
NBC news G16B1-Qp=48	28.902%	
NBC news G16B3-Qp=10	4.9427%	1.8142%
NBC news G16B3-Qp=28	28.0646%	10.3825%
NBC news G16B3-Qp=48	23.0409%	13.0418%
NBC news G16B7-Qp=10	4.53048%	1.8779%
NBC news G16B7-Qp=28	20.279%	7.7958%
NBC news G16B7-Qp=48	19.534%	10.2638%
NBC news G16B15-Qp=10	3.7882%	1.7252%
NBC news G16B15-Qp=28	14.7639%	6.7871%
NBC news G16B15-Qp=48	16.0115%	8.7637%
Star Wars IV G16B1-Qp=10	46.3945%	6.1892%
Star Wars IV G16B1-Qp=28	47.147%	11.3722%
Star Wars IV G16B1-Qp=48	18.237%	11.8242%
Star Wars IV G16B3-Qp=10	21.9308%	6.0153%
Star Wars IV G16B3-Qp=28	29.2029%	9.5832%
Star Wars IV G16B3-Qp=48	14.4279%	9.5023%
Star Wars IV G16B7-Qp=10	13.341%	5.2757%
Star Wars IV G16B7-Qp=28	19.0677%	9.8133%
Star Wars IV G16B7-Qp=48	12.5749%	6.4734%
Star Wars IV G16B15-Qp=10	8.959%	4.0796%
Star Wars IV G16B15-Qp=28	12.876%	8.2474%
Star Wars IV G16B15-Qp=48	10.514%	4.1933%

procedure. Thus, using the Markovian technique explained, we produce as many B-frames as the testing set contains, using the saved parameters and we compare them with the real values and compute the new RPE values. We used a random 80% of our data in order to train our model and the rest 20% for testing. The change in the results presented in Tables 5.10-5.11 was small, ranging from 0.5%-5%. On average, the RPE when using “Repeated random sub-sampling validation” is 2.64% bigger for H.265 traces and 1.438% bigger for H.264 traces.

Table 5.10

RPE results after “Repeated random sub-sampling validation” for H.265 Video Traces

H.265	RPE Markovian Model	RPE ”Repeated random sub-sampling validation”
BigBuckBunny_10	11.359%	14.873%
BigBuckBunny_25	11.9875%	16.531%
BigBuckBunny_40	14.1404%	19.183%
BluePlanet_10	4.3013%	5.329%
BluePlanet_25	14.018%	17.623%
BluePlanet_40	16.9615%	18.698%
Speed_10	2.5218%	3.074%
Speed_25	4.4298%	7.336%
Speed_40	4.7787%	7.601%
LakeHouse_10	3.9638%	4.280%
LakeHouse_25	5.7637%	8.499%
LakeHouse_40	6.6876%	8.912%
FindingNeverland_10	2.7957%	2.817%
FindingNeverland_25	6.1285%	9.405%
FindingNeverland_40	6.6824%	9.871%
ElephantsDream_10	17.1325%	22.353%
ElephantsDream_25	23.2173%	25.992%
ElephantsDream_40	16.3487%	18.003%
TearsOfSteel_10	8.7376%	11.294%
TearsOfSteel_25	24.545%	28.625%
TearsOfSteel_40	13.8142%	18.198%
HarryPotter_10	2.5480%	3.020%
HarryPotter_25	6.7253%	10.117%
HarryPotter_40	8.7806%	10.246%

Furthermore, we have implemented our proposed model using “Repeated random sub-sampling validation”, for the MPEG-4 movies traces used in [14], for comparison reasons. The results are presented in Table 5.12. It is clear that there is a significant reduce in the values of the RPE, proving that the markovian model works very well for the MPEG-4 compression standard.

Table 5.11

RPE results after “Repeated random sub-sampling validation” for H.264 Video Traces

H.264	RPE Markovian Model	RPE “Repeated random sub-sampling validation”
Silence of the Lambs G16B1-Qp=10	10,625%	11.724 %
Silence of the Lambs G16B1-Qp=28	28.6345%	29.429 %
Silence of the Lambs G16B1-Qp=48		
Silence of the Lambs G16B3-Qp=10	8.8341%	14.812 %
Silence of the Lambs G16B3-Qp=28	19.4561%	19.553 %
Silence of the Lambs G16B3-Qp=48	11.6337%	11.348 %
Silence of the Lambs G16B7-Qp=10	7.0562%	8.572 %
Silence of the Lambs G16B7-Qp=28	15.7460%	19.128 %
Silence of the Lambs G16B7-Qp=48	11.1072%	11.942 %
Silence of the Lambs G16B15-Qp=10	5.0496%	7.471 %
Silence of the Lambs G16B15-Qp=28	14.7654%	14.911 %
Silence of the Lambs G16B15-Qp=48	7.0128%	8.881 %
NBC news G16B1-Qp=10	1.9479%	2.354 %
NBC news G16B1-Qp=28	11.3712%	11.623 %
NBC news G16B1-Qp=48		
NBC news G16B3-Qp=10	1.8142%	2.301 %
NBC news G16B3-Qp=28	10.3825%	11.732 %
NBC news G16B3-Qp=48	13.0418%	15.071 %
NBC news G16B7-Qp=10	1.8779%	2.541 %
NBC news G16B7-Qp=28	7.7958%	9.9359 %
NBC news G16B7-Qp=48	10.2638%	11.574 %
NBC news G16B15-Qp=10	1.7252%	2.529 %
NBC news G16B15-Qp=28	6.7871%	8.005 %
NBC news G16B15-Qp=48	8.7637%	10.211 %
Star Wars IV G16B1-Qp=10	6.1892%	9.7431 %
Star Wars IV G16B1-Qp=28	14.5170%	12.632 %
Star Wars IV G16B1-Qp=48	11.8242%	12.125 %
Star Wars IV G16B3-Qp=10	6.0153%	11.093 %
Star Wars IV G16B3-Qp=28	9.5832%	13.103 %
Star Wars IV G16B3-Qp=48	9.5023%	10.326 %
Star Wars IV G16B7-Qp=10	5.2757%	6.029 %
Star Wars IV G16B7-Qp=28	9.8133%	9.927 %
Star Wars IV G16B7-Qp=48	6.4734%	8.984 %
Star Wars IV G16B15-Qp=10	4.0796%	5.551 %
Star Wars IV G16B15-Qp=28	8.2474%	8.9612 %
Star Wars IV G16B15-Qp=48	4.1933%	6.206 %

Table 5.12
Markov Model- RPE results for MPEG-4 Video Traces

	RPE Lanfranchi,Bing	RPE our Markov Model ("Repeated random sub-sampling validation")
Silence of the Lambs		
HQ	7.47%	5.98%
MQ	12.46%	8.56%
LQ	17.10%	11.47%
Aladdin		
HQ	22.40%	4.24%
MQ	26.06%	4.56%
LQ	28.70%	5.52%
Ski		
HQ	7.09%	2.61%
Simpsons		
HQ	10.78%	6.18%

5.5 Comparison

5.5.1 With LSTAR Model

Additionally to the comparison of our model with [14], which is the core of this work, we have also compared the results of our model with two other models. The first is a very recently proposed model for video traffic prediction [1]. In [1] the authors propose the use of a non-linear autoregressive model (Logistic Smooth Transition Autoregressive, LSTAR) to predict video traffic, and they introduce a number of adaptive algorithms to obtain the parameters of LSTAR.

A direct comparison between our results and the results in [1] is not possible because we only focus on B -frames, whereas the authors in [1] try to model the whole trace. However, the Normalized Mean Square Error (NMSE) values for both models can serve as an indication of the efficiency of our modeling approach. The NMSE is defined as [43]:

$$NMSE = \frac{1}{\sigma^2} \frac{1}{N} \sum_{n=1}^N (x - \bar{x})^2 \quad (5.14)$$

where x is the real size, \bar{x} is the predicted size, σ^2 is the variance of x over N B-frames.

Table 5.13 presents the NMSE results of our model, for B -frames, against the best results presented in [1], for the whole trace. It is clear that our model achieves a very low NMSE for all 4 traces, much lower than the one achieved in [1] for the whole trace.

Table 5.13
NMSE Comparison with [1]

NMSE	Harry Potter H.265 QP=10	Harry Potter H.265 QP=35	NBC News H.264 G16B3 QP=10	NBC News H.264 G16B3 QP=34
LSTAR Model (for the whole trace)	0.1491	0.1769	0.3217	0.3186
Markovian model (for B-frames)	0.0061	0.0822	0.0077	0.1113

Figure 5.2 and 5.3 present indicatively the Q-Q plots when our model is implemented on the two Harry Potter traces and two NBC news traces, for two random B-frames each times. The Q-Q plots show that the predicted traffic matches closely with the original traffic from the traces.

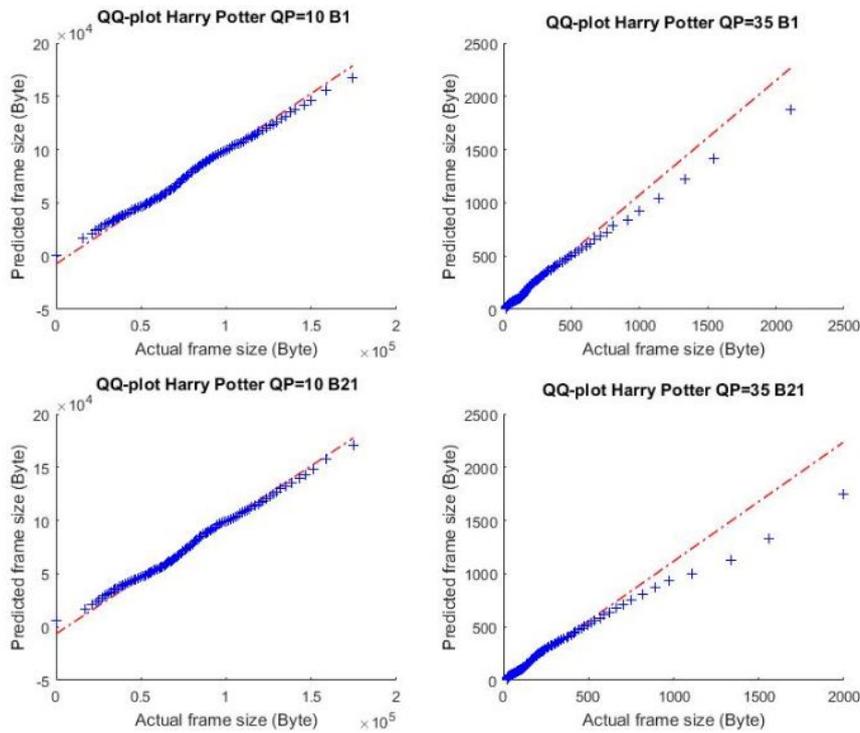


Fig. 5.2 Markovian Model – QQ plot results for the Harry Potter QP=10 and QP=35 traces

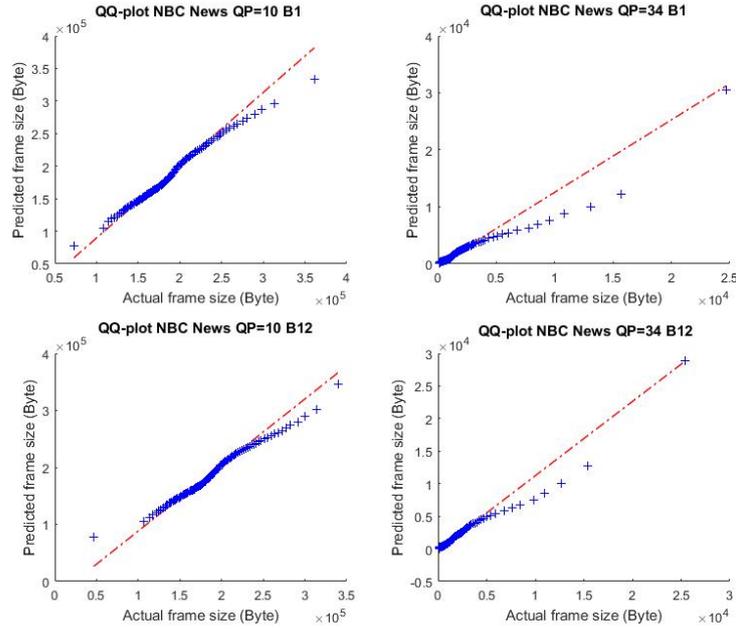


Fig. 5.3 Markovian Model – QQ plot results for the NBC news QP=10 and QP=35 traces

5.5.2 With Markov-modulated Gamma-based model

The second model from the literature that we compared our model with is the work in [2], which proposes the use of a Markov-modulated Gamma-based framework for modeling full videos. A direct comparison between our results and the results in [16] is possible because the authors' model generates the sizes of *I*-frames, *P*-frames and *B*-frames separately.

The work in [2] demands, for its highly accurate model, the following steps:

- formation of clips which are consecutive sequences of GoPs in order to group similar size GoPs
- the use of clips for the formation of shot classes which are unions of distinct, but not necessarily consecutive clips
- the removal from the trace of the 1% of very high and very low traces in order to compute an offset that is used for modeling frame sizes based on a shifted Gamma distribution
- the formation of video segments (maximal consecutive sequences of clips belonging in the same shot class)

- the calculation of the probabilities for inter-shot class transitions.

The work in [2] proposed two variations of the proposed model, Model A and Model B, which as explained however by the authors lead practically to identical video frame-length distributions. Therefore, we have used Model A to compare our model with. Hence, we implemented the steps explained above with a slight difference; at the end of the procedure we only generated the sizes of the B-frames. We computed the RPE for every video trace both for H.265 and H.264 compression standards.

Table 5.14
RPE comparison with [2] for H.265 Video Traces

H.265	RPE Markovian Model	RPE Markov-Modulated Gamma-Based Framework
Harry Potter QP=10	2.5480%	2.3900%
Harry Potter QP=25	6.7253%	6.0772%
Harry Potter QP=40	8.7806%	7.4380%
Elephant QP=10	17.1325%	15.490%
Elephant QP=25	23.2173%	18.042%
Elephant QP=40	16.3487%	15.730%
FindingNeverland P=10	2.7957%	3.244%
FindingNeverland P=25	6.1285%	6.0135%
FindingNeverland P=40	6.6824%	4.830%
BigBuckBunny P=10	11.3591%	13.4389%
BigBuckBunny P=25	11.9875%	9.8147%
BigBuckBunny P=40	14.1404%	11.219%
BluePlanet P=10	4.3013%	7.3021%
BluePlanet P=25	14.0182%	11.4502%
BluePlanet P=40	16.9615%	15.799%
Speed P=10	2.5218%	2.2501%
Speed P=25	4.4298%	7.324%
Speed P=40	4.7787%	6.619%
TearsOfSteel P=10	8.7376%	6.0931%
TearsOfSteel P=25	24.5541%	17.023%
TearsOfSteel P=40	13.8142%	12.5902%
LakeHouse P=10	3.9638%	3.2911%
LakeHouse P=25	5.7637%	9.7713%
LakeHouse P=40	6.6876%	6.6892%

The results of Table 5.15, when using the proposed model from [2], show that on average our model has a 0.5% accuracy higher than [2] for H.264 traces. The results

of Table 5.14, when using the proposed model from [2], show that on average the model from [2] has a 1.5% higher accuracy than our model for H.265 traces. However, it is important to note that the proposed model in [2] works well only for already known traces. In terms of the possibility of its real-time implementation, this would be difficult because of its complexity due to the large number of the aforementioned steps that need to be completed (new clips, shot classes and segments to be computed and formed, new offset calculations and new inter-shot class transitions for the seven shot classes) while new video frames would be arriving into the network every few tens of ms.

In terms of our work, since the number of B -frames in all H.264 and H.265 traces under study is small, the running time of our proposed algorithm is low. Indicatively, the training of the model for an individual trace needs between 30 (for most of the traces) and 80 seconds and the prediction takes 10-15 seconds. This time is quite low considering that: a) for already known traces the training will have already taken place, and b) for unknown traces which the user will want to download and for which the training needs to be done on-the-fly, an initial approximation can be used by utilizing the trace that is closer to the one being downloaded in terms of mean and standard deviation, and within a few tens of seconds our accurate model will be trained.

Table 5.15
RPE comparison with [2] for H.264 Video Traces

H.264	RPE Markovian Model	RPE Markov-Modulated Gamma-Based Framework
Silence of the Lambs G16B1-Qp=10	10.6287%	17.126
Silence of the Lambs G16B1-Qp=28	28.6345%	17.838%
Silence of the Lambs G16B1-Qp=48		
Silence of the Lambs G16B3-Qp=10	8.8341%	13.304%
Silence of the Lambs G16B3-Qp=28	19.4561%	15.189%
Silence of the Lambs G16B3-Qp=48	11.6337%	7.923%
Silence of the Lambs G16B7-Qp=10	7.0562%	9.5028%
Silence of the Lambs G16B7-Qp=28	15.7460%	15.998%
Silence of the Lambs G16B7-Qp=48	11.1072%	8.592%
Silence of the Lambs G16B15-Qp=10	5.0496%	9.4402%
Silence of the Lambs G16B15-Qp=28	14.7654%	26.007%
Silence of the Lambs G16B15-Qp=48	7.0128%	12.464%
NBC news G16B1-Qp=10	1.9479%	1.749
NBC news G16B1-Qp=28	11.3712%	7.602
NBC news G16B1-Qp=48		
NBC news G16B3-Qp=10	1.8142%	2.1235%
NBC news G16B3-Qp=28	10.3825%	9.139%
NBC news G16B3-Qp=48	13.0418%	9.836%
NBC news G16B7-Qp=10	1.8779%	1.9103%
NBC news G16B7-Qp=28	9.4949%	10.076%
NBC news G16B7-Qp=48	10.2638%	8.9023%
NBC news G16B15-Qp=10	1.7252%	2.015%
NBC news G16B15-Qp=28	6.7871%	7.9162%
NBC news G16B15-Qp=48	8.7637%	9.555%
Star Wars IV G16B1-Qp=10	6.9976%	10.331%
Star Wars IV G16B1-Qp=28	11.3722%	10.544
Star Wars IV G16B1-Qp=48	11.8242%	7.820%
Star Wars IV G16B3-Qp=10	6.0153%	8.712%
Star Wars IV G16B3-Qp=28	9.5832%	13.442%
Star Wars IV G16B3-Qp=48	9.5023%	8.2092%
Star Wars IV G16B7-Qp=10	5.2757%	9.130%
Star Wars IV G16B7-Qp=28	9.8133%	9.8230%
Star Wars IV G16B7-Qp=48	6.4734%	7.252%
Star Wars IV G16B15-Qp=10	4.0796%	7.2101%
Star Wars IV G16B15-Qp=28	8.2474%	11.502%
Star Wars IV G16B15-Qp=48	4.1933%	12.128%

Chapter 6

Conclusion and Future Work

In this work we have focused on predicting the size of the B -frames of MPEG-4, H.264 and H.265 encoded videos. An accurate prediction of B -frames' sizes can reduce video bandwidth requirements and smoothen the encoded video stream in cases of network congestion, by selective B -frame dropping. We initially implemented a prediction method from the literature which is based on linear regression using inter-frame correlation coefficients and had been shown to provide high accuracy for MPEG-4 videos. This method, and some variants of it that we explored, did not provide accurate prediction results for the H.264 and H.265 traces in our study.

For this reason, we have proposed and implemented a new prediction approach, based on simulated annealing. Our model was shown to clearly outperform the prior work from [9] in terms of accuracy. Still, it should be emphasized that even with the use of Simulated Annealing there are quite a few cases where the achieved RPE was high. The reason is that we have based our metaheuristic search on the same logic as that of [9], which is to try to perform linear regression with the best possible set of equations, in order to predict future B frames' sizes. However, while analyzing the datasets and specifically the points where the prediction was least accurate, we found that in many of the H.265 traces and in several H.264 there was a significant number of B -frames with much smaller size than the mean B -frame (the problem was not encountered as often in the MPEG-4 traces, and this is the reason for the better results of both approaches for MPEG-4). These very small B -frames were randomly placed in the video trace, and due to their very small size they couldn't be well predicted using the equations of the linear regression approach.

Hence, we have proposed and implemented a new prediction approach, based on splitting the B -frames in two subsets (big/small) based on the frames' sizes, finding the best distribution fit for each subset's population of sizes, and using a Markovian model

to simulate the transitions between the subsets. Our model was shown to significantly outperform the prior work from [14], offering an average improvement of over 50% in terms of accuracy, as well as achieving less than half the Normalized Mean Square Error achieved (for the whole trace) by another competent model in [14]. Our model was also shown to have comparable accuracy with another Markovian model from the literature [2], which however is quite more complex and therefore difficult to implement for unknown traces where the computations would need to be done on-the-fly. The accuracy of our model comes at the cost of only a few tens of seconds, which is the running time of our algorithm for unknown traces.

In future work we intend to implement our model on a wider variety of traces encoded with H.264 and H.265 traces. We also intend to use simple but efficient machine learning techniques, like clustering algorithms, for predicting B -frames' sizes and to compare the efficiency of these techniques against our model, both in terms of accuracy and in terms of computational complexity.

References

- [1] H. Kalbkhani, M. G. Shayesteh, and N. Haghghat, "Adaptive Istar model for long-range variable bit rate video traffic prediction," *IEEE Transactions on Multimedia*, 2016.
- [2] U. K. Sarkar, S. Ramakrishnan, and D. Sarkar, "Modeling full-length video using markov-modulated gamma-based framework," *IEEE/ACM transactions on Networking*, vol. 11, no. 4, pp. 638–649, 2003.
- [3] Cisco virtual networking index: Forecast and methodology, 2015-2020. <http://www.cisco.com/c/dam/en/us/solutions/collateral/service-provider/visual-networking-index-vni/complete-white-paper-c11-481360.pdf>.
- [4] D. Marpe, T. Wiegand, and G. J. Sullivan, "The h. 264/mpeg4 advanced video coding standard and its applications," *IEEE communications magazine*, vol. 44, no. 8, pp. 134–143, 2006.
- [5] G. J. Sullivan, J. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (hevc) standard," *IEEE Transactions on circuits and systems for video technology*, vol. 22, no. 12, pp. 1649–1668, 2012.
- [6] Pixeltools, White paper [Online]. http://www.pixeltools.com/rate_control_paper.html.
- [7] X. Qi, Q. Yang, D. T. Nguyen, G. Peng, G. Zhou, B. Dai, D. Zhang, and Y. Li, "A context-aware framework for reducing bandwidth usage of mobile video chats," *IEEE Transactions on Multimedia*, vol. 18, no. 8, pp. 1640–1649, 2016.
- [8] K. Miller, D. Bethanabhotla, G. Caire, and A. Wolisz, "A control-theoretic approach to adaptive video streaming in dense wireless networks," *IEEE Transactions on Multimedia*, vol. 17, no. 8, pp. 1309–1322, 2015.
- [9] Q. M. Qadir, A. A. Kist, and Z. Zhang, "A novel traffic rate measurement algorithm for quality of experience-aware video admission control," *IEEE Transactions on Multimedia*, vol. 17, no. 5, pp. 711–722, 2015.
- [10] Cisco video and telepresence architecture design guide. http://www.cisco.com/c/en/us/td/docs/voice_ip_comm/uc_system/design/guides/videodg/vidguide/qos.html.
- [11] Y. Li, Z. Li, M. Chiang, and A. R. Calderbank, "Content-aware distortion-fair video streaming in congested networks," *IEEE Transactions on Multimedia*, vol. 11, no. 6, pp. 1182–1193, 2009.

-
- [12] J. Greengrass, J. Evans, and A. C. Begen, "Not all packets are equal, part i: Streaming video coding and sla requirements," *IEEE Internet Computing*, vol. 13, no. 1, pp. 70–75, 2009.
- [13] Y. Xiao, Y. Zhang, M. Nolen, J. H. Deng, and J. Zhang, "A cross-layer approach for prioritized frame transmissions of mpeg-4 over the ieee 802.11 and ieee 802.11 e wireless local area networks," *IEEE systems journal*, vol. 5, no. 4, pp. 474–485, 2011.
- [14] L. I. Lanfranchi and B. K. Bing, "Mpeg-4 bandwidth prediction for broadband cable networks," *IEEE Transactions on Broadcasting*, vol. 54, no. 4, pp. 741–751, 2008.
- [15] A. Kalampogia, "Video traffic modeling by exploiting inter-frame correlation coefficients", diploma thesis, Electrical and Computer Engineering, 2014.
- [16] D. M. Lucantoni, M. F. Neuts, and A. R. Reibman, "Methods for performance evaluation of vbr video traffic models," *IEEE/ACM Transactions On Networking*, vol. 2, no. 2, pp. 176–180, 1994.
- [17] A. M. Dawood and M. Ghanbari, "Content-based mpeg video traffic modeling," *IEEE Transactions on Multimedia*, vol. 1, no. 1, pp. 77–87, 1999.
- [18] K. Chandra and A. R. Reibman, "Modeling one-and two-layer variable bit rate video," *IEEE/ACM Transactions on Networking (TON)*, vol. 7, no. 3, pp. 398–413, 1999.
- [19] Q. Ren and H. Kobayashi, "Diffusion approximation modeling for markov modulated bursty traffic and its applications to bandwidth allocation in atm networks," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 5, pp. 679–691, 1998.
- [20] M. Dai, Y. Zhang, and D. Loguinov, "A unified traffic model for mpeg-4 and h. 264 video traces," *IEEE Transactions on Multimedia*, vol. 11, no. 5, pp. 1010–1023, 2009.
- [21] A. Bhattacharya, A. G. Parlos, and A. F. Atiya, "Prediction of mpeg-coded video source traffic using recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 51, no. 8, pp. 2177–2190, 2003.
- [22] S. Tanwir and H. Perros, "A survey of vbr video traffic models," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 4, pp. 1778–1802, 2013.
- [23] A. K. Al Tamimi, R. Jain, and C. So-In, "Modeling and generation of avc and svc-ts mobile video traces for broadband access networks," in *Proceedings of the First Annual ACM SIGMM Conference on Multimedia Systems*, ser. MMSys '10. New York, NY, USA: ACM, 2010, pp. 89–98. [Online]. Available: <http://doi.acm.org/10.1145/1730836.1730848>
- [24] D. P. Heyman, "The gbar source model for vbr videoconferences," *IEEE/ACM transactions on networking*, vol. 5, no. 4, pp. 554–560, 1997.

- [25] M. Frey and S. Nguyen-Quang, "A gamma-based framework for modeling variable-rate mpeg video sources: the gop gbar model," *IEEE/ACM transactions on networking*, vol. 8, no. 6, pp. 710–719, 2000.
- [26] D. P. Heyman, A. Tabatabai, and T. Lakshman, "Statistical analysis and simulation study of video teleconference traffic in atm networks," *IEEE Transactions on circuits and systems for video technology*, vol. 2, no. 1, pp. 49–59, 1992.
- [27] D. Heyman, T. Lakshman, A. Tabatabai, and H. Heeke, "Modeling teleconference traffic from vbr video coders," in *Communications, 1994. ICC'94, SUPERCOMM/ICC'94, Conference Record, 'Serving Humanity Through Communications. IEEE International Conference on.* IEEE, 1994, pp. 1744–1748.
- [28] S. Xu and Z. Huang, "A gamma autoregressive video model on atm networks," *IEEE Transactions on circuits and systems for video technology*, vol. 8, no. 2, pp. 138–142, 1998.
- [29] A. Lazaris, P. Koutsakis, and M. Paterakis, "A new model for video traffic originating from multiplexed mpeg-4 videoconference streams," *Performance Evaluation*, vol. 65, no. 1, pp. 51–70, 2008.
- [30] I. Spanou, A. Lazaris, and P. Koutsakis, "Scene change detection-based discrete autoregressive modeling for mpeg-4 video traffic," in *Communications (ICC), 2013 IEEE International Conference on.* IEEE, 2013, pp. 2386–2390.
- [31] A. M. Kholaf, T. D. Todd, P. Koutsakis, and A. Lazaris, "Energy efficient h. 263 video transmission in power saving wireless lan infrastructure," *IEEE Transactions on Multimedia*, vol. 12, no. 2, pp. 142–153, 2010.
- [32] S. Colonnese, S. Russo, F. Cuomo, T. Melodia, and I. Rubin, "Timely delivery versus bandwidth allocation for dash-based video streaming over lte," *IEEE Communications Letters*, vol. 20, no. 3, pp. 586–589, 2016.
- [33] I. Sodagar, "The mpeg-dash standard for multimedia streaming over the internet," *IEEE MultiMedia*, vol. 18, no. 4, pp. 62–67, 2011.
- [34] N. Bouzakaria, C. Concolato, and J. Le Feuvre, "Fast dash bootstrap," in *Multimedia Signal Processing (MMSP), 2015 IEEE 17th International Workshop on.* IEEE, 2015, pp. 1–6.
- [35] Trace Files and Statistics. <http://trace.eas.asu.edu/tracemain.html>.
- [36] G. Van der Auwera, P. David, and M. Reisslein, "Traffic and quality characterization of single-layer video streams encoded with the h," 2008.
- [37] Mpeg-4 and h.263 video traces for network performance evaluation. <http://www-tnk.ee.tu-berlin.de/research/trace/trace.html>.
- [38] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by Simulated Annealing," *science*, vol. 220, no. 4598, pp. 671–680, 1983.

-
- [39] The Simulated Annealing Algorithm, [Online]. <http://katrinaeg.com/simulated-annealing.html>.
- [40] R. Bai, J. Blazewicz, E. K. Burke, G. Kendall, and B. McCollum, "A Simulated Annealing Hyper-Heuristic methodology for flexible decision support," *4OR: A Quarterly Journal of Operations Research*, vol. 10, no. 1, pp. 43–66, 2012.
- [41] P. B. Hansen. (1992) Simulated annealing. http://surface.syr.edu/cgi/viewcontent.cgi?article=1158&context=eecs_techreports.
- [42] A. M. Law, W. D. Kelton, and W. D. Kelton, *Simulation modeling and analysis*. McGraw-Hill New York, 1991, vol. 2.
- [43] Y. Liang, "Real-time vbr video traffic prediction for dynamic bandwidth allocation," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 34, no. 1, pp. 32–47, 2004.