# Macroblock Level Rate Control for Low Delay H.264/AVC based Video Communication

Min Gao *, Burak Cizmeci †, Michael Eiler †, Eckehard Steinbach †, Debin Zhao *, Wen Gao *
* Department of Computer Science and Technology
Harbin Institute of Technology, Harbin, China
† Chair of Media Technology
Technical University of Munich, Munich, Germany

***Abstract*—In this paper, we propose a macro-block (MB) level rate control algorithm for low delay H.264/AVC video communication based on the $\rho$ domain rate model. In the proposed algorithm, an exponential model is used to characterize the relation between $\rho$ and the quantization step ($Qstep$) at the MB level, with which the quantization parameter ($QP$) for a MB can be obtained. Furthermore, a switched $QP$ calculation scheme is introduced to obtain the $QP$ for each MB to avoid large deviation of the actual frame size from the target bit budget. Compared with the original $\rho$ domain rate control, the proposed method can achieve better video quality and improved bit-rate accuracy. Meanwhile, the computational complexity is also significantly reduced.**

## I. Introduction

In many video communication systems, the compressed video bit-stream needs to be transmitted over a constant bit rate (CBR) channel. The output bit rate of a typical video encoder is, however, variable and different from one frame to another due to the diversity of the frame content. In order to address this problem, a buffer is required to handle the variable bit-rate (VBR) video stream at the transmitter/receiver. A rate control scheme has to be employed to adjust the coding parameters to prevent buffer overflow and underflow.

For conversational and interactive video applications, very stringent end-to-end delay (from capture to display) constraints apply, which in turn limits the buffer size that can be used. Because small buffers easily cause overflow and underflow, more accurate rate control algorithms are required for low delay video communication. In general, rate control can be applied at the frame level or even at the macro-block (MB) level. Compared with frame level rate control, MB level rate control leads to lower coding efficiency, but can achieve more accurate target bit budget matching and improved buffer regulation. In this paper, we propose a simple, yet accurate MB level rate control algorithm for low delay communication of H.264/AVC encoded video based on the $\rho$ domain rate model [4].

Several MB level rate control algorithms for H.264/AVC have been introduced in the literature. In [1], a rate control algorithm is proposed which employs a quadratic rate-quantization (R-Q) model, and which was adopted in the H.264/AVC reference software. To improve the performance of [1], Jiang et al. developed more accurate frame level bit allocation and mean absolute difference (MAD) estimation in [2]. To improve the model parameter estimation accuracy, a linear R-Q model based MB level rate control was proposed in [3] using a context adaptive prediction scheme. However, these algorithms suffer from occasional large errors in the bit-rate estimation due to the inaccurate source models and hence require a larger buffer size.

It was found in [4] that the bit-rate ($R$) shows a linear relationship with $\rho$, which is defined as the percentage of zero transform coefficients after quantization. This linear model between $R$ and $\rho$ has been exploited for rate control in H.263 and MPEG4 [5, 6], and can produce more accurate bit-rate estimation. Afterwards, a $\rho$ domain rate control scheme was proposed for H.264/AVC in [7] with a two-loop encoding pipeline, in which the frame level statistics are collected in the first loop and used in the second loop to determine the proper $QP$ for each MB. An improved $\rho$ domain rate control was proposed in [8] with a more accurate header bits estimation.

However, it is not easy to find a one-to-one mapping between $\rho$ and $QP$ due to the complicated coefficient quantization scheme in H.264/AVC [9]. In [7] and [8], the transform coefficients are quantized using all possible $QP$s to obtain the $(\rho, QP)$ table. Then, the $(\rho, QP)$ pairs in this table are searched to find the proper $QP$ for a given $\rho$. The high complexity of this process makes it impractical for low delay video communication. To reduce the complexity, a linear model was proposed in [10] to establish the relationship among $Qstep$, frame complexity (represented by MAD) and $\rho$, and a frame level rate control was proposed for scalable video coding based on the model. However, this model is not accurate enough at the MB level, and may induce large errors in the bit-rate estimation. Therefore, it is required to develop an accurate and low complexity MB level rate control algorithm that maintains the high accuracy of the $\rho$ domain model in bit-rate estimation.

In this paper, we propose a MB level rate control algorithm based on the $\rho$ domain rate model for low delay video communication. In this algorithm, an exponential model is adopted to characterize the relation between $\rho$ and $Qstep$ at the MB level. Furthermore, a switched $QP$ calculation scheme is introduced to avoid large deviations of the frame size from the target bit-rate. In the proposed scheme, the $QP$ is calculated from the exponential model if the remaining bit budget is larger than a threshold; otherwise the $QP$ of the previous MB plus a constant is used as the $QP$ of the current MB.

The rest of the paper is organized as follows. Section II gives an overview of the original $\rho$ domain rate control. Section III describes the proposed exponential model for the relation between $\rho$ and $Qstep$. Section IV presents the proposed MB level rate control scheme in detail. The experimental results are presented in Section V. Finally, Section VI concludes the paper.

## II. Review of $\rho$ Domain Rate Control

In [4], He et al. introduced a linear rate model for transform coding of images and videos:

$$R = \theta \cdot (1 - \rho) \quad (1)$$

where $R$ is the output texture bits, $\rho$ is the percentage of transform coefficients which become zero after quantization, and $\theta$ is a constant slope parameter that is closely related to the frame content. This linear model allows for accurate bit rate estimation. The high accuracy is also maintained at the MB level, as illustrated in Fig. 1.

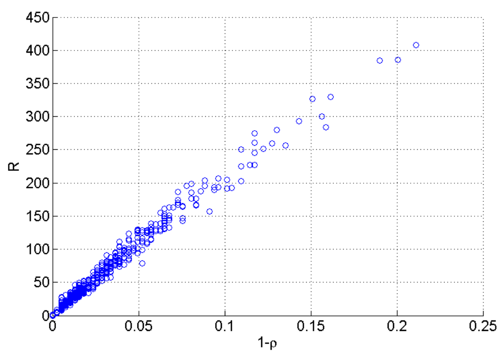


Fig. 1. Relationship between $R$ and $1 - \rho$ at the MB level for the Foreman video sequence (*CIF*, encoded with x264 at 400kbps).

The fraction of zeros $\rho$ increases for growing $QP$. This implies that there is a one-to-one mapping between $\rho$ and $QP$ [7], which is shown as follows:

$$\rho(QP) = \frac{1}{S} \sum_{|x|<\Delta} P(x) \quad (2)$$

where $P(x)$ is the distribution of the un-quantized transform coefficients, $S$ is the total number of transform coefficients in the frame and $\Delta$ is the dead zone of the quantizer that is determined by $QP$.

The rate control for H.264/AVC in [7] is implemented as follows based on (1) and (2):

1. Collecting frame level statistics:
   Perform motion compensation, intra prediction and block transform for all MBs in the current frame. Find the distribution of transform coefficients $P(x)$.
2. Determine $QP$ for the current MB:
   Determine the target fraction of zeros for the remaining MBs according to the remaining bit budget $R_{left}$ using (1). Based on the one-to-one mapping between $\rho$ and $QP$ in (2), determine the $QP$ for the current MB.
3. Parameter updating:
   Encode the current MB with the obtained $QP$. Update $\theta$ in (1) with the fraction of zero coefficients and the number of bits produced by the current MB. Update $P(x)$ by removing the transform coefficients in the current MB from the distribution $P(x)$.
4. Loop:
   Repeat step 2 and step 3 until all MBs in the frame are encoded.

## III. The Exponential Model for $\rho$ and $QP$

Although (2) provides a method to estimate $QP$ from the calculated $\rho$, it is not easy to find the one-to-one mapping between $\rho$ and $QP$ due to the complicated quantization process in H.264/AVC. For example, in [7] and [8], the transform coefficients for all MBs in a frame are quantized with all possible $QP$ values to obtain the $(\rho, QP)$ table. Then, all $(\rho, QP)$ pairs in the table are searched to get the proper $QP$ for a given $\rho$. This process of determining $QP$ is obviously computationally very demanding.

Hence, an efficient model that captures the relationship between $\rho$ and $QP$ is required. Our experiments show that the relationship between $\rho$ and $Qstep$ can be modeled using an exponential function:

$$\rho = 1 - a \cdot e^{b \cdot Qstep} \quad (3)$$

where $a$ and $b$ are the model parameters. The relationship between $Qstep$ and $QP$ is shown in (4).

$$Qstep = 2^{\frac{QP-4}{6}} \quad (4)$$

Fig. 2 shows that the proposed model has an excellent estimation accuracy. Table I provides the correlation coefficients between the actual value and the estimated ones for the selected test sequences. It can be seen that the correlation between the actual data and the estimated one is greater than 0.9, which indicates that the exponential model accurately estimates $QP$ from $\rho$.

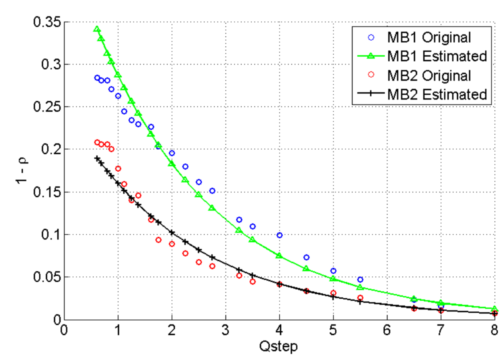


Fig. 2. Relationship between $1 - \rho$ and $Qstep$ at the MB level for the Foreman test video sequence (*CIF*, encoded with x264 at 400kbps).

TABLE I
CORRELATION COEFFICIENTS BETWEEN THE ACTUAL VALUE AND THE ESTIMATED ONES

| Sequences | Bit rate (kbps) | Correlation coefficient |
|---|---|---|
| Football | 400 | 0.917 |
| | 1000 | 0.983 |
| Foreman | 400 | 0.970 |
| | 1000 | 0.951 |
| Mobile | 400 | 0.969 |
| | 1000 | 0.981 |

To perform rate control based on (3), the parameters $a$ and $b$ need to be estimated first. After performing RDO (rate distortion optimization), for each MB, the non-quantized transform coefficients under the best mode are quantized with $QP_1$ and $QP_2$. Then, for each MB, $\rho_1$ and $\rho_2$ are calculated for $QP_1$ and $QP_2$, respectively. So the parameters $a$ and $b$ are estimated with (3) using $\rho_1$, $\rho_2$ and $Qstep$ corresponding to $QP_1$ and $QP_2$.

When the parameters $a$ and $b$ are available, the $Qstep$ can be computed using (3) for a given $\rho$. Then the corresponding $QP$ is obtained from $Qstep$ using (4).

## IV. THE PROPOSED MACROBLOCK LEVEL RATE CONTROL ALGORITHM

The objective of rate control is to provide the best possible video quality for a constraint bit budget, which can be achieved by MB level bit allocation and accurate $QP$ selection. In this section, we first present the bit allocation algorithm at frame and MB level; then a switched $QP$ calculation scheme at the MB level is described; finally, the whole rate control algorithm is summarized.

### A. Bit allocation at frame and MB level

Since the buffer size is very small in low-delay video communication, a constant bit budget per frame is assumed in this paper:

$$R_T = \frac{R_C}{F} \tag{5}$$

where $R_C$ is the transmission rate of the CBR channel, $F$ is the frame rate of the video sequence, and $R_T$ is the frame level bit budget.

After the determination of $R_T$ , the next step is to distribute the bit budget $R_T$ among the MBs in a frame to minimize the frame distortion. Since the MB level rate control works sequentially through these MBs, it is generally observed that the actual number of bits generated for a frame is typically larger than the target bit budget. This implies that the bit budget will be used up before encoding all MBs. To be fair to the MBs near the end of a frame, the MB level bit allocation proposed in [2] is used here, which is shown by (6).

$$R_{MB}^i = (\omega_1 \cdot \frac{R_{left}}{N_{left}} + \omega_2 \cdot avg\_R_{MB}) \cdot \frac{MAD_i}{MAD_F} \cdot S_i \tag{6}$$

where $R_{MB}^i$ is the assigned number of bits for $MB_i$ (the MB at position $i$); $R_{left}$ and $N_{left}$ are the number of remaining bits and the number of the un-coded MBs in a frame, respectively; $avg\_R_{MB}$ is the average target number of bits for each MB, which is given by (7);

$$avg\_R_{MB} = \frac{R_T}{N_{MB}} \tag{7}$$

where $N_{MB}$ is the number of MBs in a frame. $MAD_i$ is the MAD of $MB_i$; $MAD_F$ is the MAD of the current frame; $S_i$ is a position-dependent scaling factor, which is given by (8):

$$S_i = \alpha_0 \cdot \frac{i}{N_{MB}} + \alpha_1 \tag{8}$$

where $\alpha_0$ and $\alpha_1$ are constants, which are set as 0.4 and 0.8, respectively.

In [2], the weighting factors $\omega_1$ and $\omega_2$ in (6) are set as 0.7 and 0.3, respectively. The frame size produced under these values is larger than the target number of bits. To avoid the large deviation of the frame size from the target bits, $\omega_1$ and $\omega_2$ are set as 0.2 and 0.008 according to our experiments.

Given the allocated bits $R_{MB}^i$ for $MB_i$, the number of texture bits $tex\_R_{MB}^i$ for this MB is given by:

$$tex\_R_{MB}^i = R_{MB}^i - R_{hdr}^i \tag{9}$$

where $R_{hdr}^i$ is the estimated number of header bits for $MB_i$, which is the average number of header bits generated by all previously coded MBs in the current frame.

### B. QP determination at the MB level

The percentage of zero coefficients $\rho_i$ among the quantized transform coefficients in $MB_i$ *is* calculated using (1) when $tex\_R_{MB}^i$ is available. The quantization step $Qstep_i$ for $MB_i$ is then computed using (3). Finally, the corresponding $QP_i$ is obtained with (4). To maintain the quality smoothness within a frame, the $QP_i$ should be limited within a range. In this paper, the $QP$ adjustment scheme proposed in [2] is adopted, shown as follows:

$$QP_i = min\{QP_{i-1} + \Delta QP, max\{QP_i, QP_{i-1} - \Delta QP\} \tag{10}$$

where $QP_{i-1}$ is the $QP$ of $MB_{i-1}$, and $\Delta QP$ is the varying range of $QP$ along MBs. The initial value of $\Delta QP$ is 2, and it is updated as follows after encoding each MB i.e., $MB_j$:

$$\Delta QP = \begin{cases} 1, & if\ QP_j \geq 25 \\ 2, & otherwise \end{cases} \tag{11}$$

Since the buffer size is very small in low delay video communication, we introduce a threshold to control the $QP$ calculation to avoid large deviation of the frame size from the target bit budget, which is defined as:

$$thr = n \cdot \frac{prev\_R_{hdr}}{prev\_R_{total}} \cdot R_{left} \tag{12}$$

where $n$ is a constant, $prev\_R_{hdr}$ and $prev\_R_{total}$ are the header bits and the total bits produced by the previous frame, respectively; $R_{left}$ is the remaining bits for the uncoded MBs in the current frame.

Therefore, a switched calculation of $QP$ is described as follows:

**If** $R_{left} \geqslant thr$ **then**

$QP$ is calculated with (3), (4) and (10).

**else**

$QP$ is set as $QP_{i-1} + 4$.

**End If**

### C. Summary of the proposed rate control algorithm

We propose a two stage rate control algorithm. In the first stage, the motion estimation and mode decision are performed. We record the MVs (Motion Vector), prediction difference and MAD for the best mode of each MB. In the second stage, the proposed rate control algorithm is used to get the final $QP$ for each MB, and then the actual encoding is performed. Although the proposed rate control is two-stage, the motion estimation and mode decision are performed only once. So it has a similar computational complexity as one pass rate control algorithms. The detail description of the proposed rate control algorithm is described as follows:

1. Frame level bit budget:
   The frame level bit budget is computed using (5).
2. The first stage: rate distortion optimization (RDO):
   a Determination of the initial $QP$ used for RDO:
   **If** the current frame is an Intra frame, **then**

$$QP_{init} = \begin{cases} 30, & \textit{If bpp} \geqslant 0.13 \\ 45, & Otherwise \end{cases}$$

   where *bpp* denotes the bits per pixel.
   **else**
   the average $QP$ of the previous frame is used.
   **End If**

   b Perform RDO for each MB:
   Motion estimation and mode decision are conducted for all MBs in the current frame using the initial $QP$. Then the MVs, prediction difference and MAD for the best mode of each MB are recorded.
3. The second stage: actual encoding stage:
   (a) Bit allocation for individual $MB_j$:
   Get the texture bits for the MB with the scheme in Subsection IV-A.
   (b) Calculation of the model parameters for the MB:
   Calculate the parameters $a$ and $b$ in (3) with the methods in Section III.
   (c) Final $QP$ calculation for the MB:
   Compute the final $QP$ of the MB with the scheme in Subsection IV-B.
   (d) Perform actual encoding for the MB:
   Encoding the mode, MV and quantized transform coefficients with the final $QP$.
   (e) Update model parameter $\theta$ in (1):
   After encoding the MB, the value of $\theta$ is updated with the following equation:

$$\theta = \frac{R_m}{384 \cdot N_m - N_{zero}}$$

   where $N_m$ is the number of coded MBs in the current frame, $R_m$ is the number of bits produced by these coded MBs, and $N_{zero}$ is the number of zero coefficients in these coded MBs. Note that there are 384 coefficients for a MB in YUV 4:2:0 format.
4. Loop:
   Repeat step 3 until all MBs in the frame are coded.

## V. Experimental results

The proposed rate control scheme is implemented in x264. The encoder is configured to conform to the baseline profile. CAVLC is used for entropy coding, and there is only one reference frame for each prediction frame. We select the $CIF$ format sequences $Bus$, $Container$, $Football$, $Foreman$ and $Mobile$ as a test set, whose frame rate are all 25 *fps*. These five test sequences are selected since they are representative of different levels of spatial and temporal complexity. For each sequence, 250 frames are encoded, in which the first frame is encoded as I frame and the remaining frames are encoded as P frames.

We compare the proposed rate control algorithm with the original $\rho$ domain rate control algorithm in [7], in which the transform coefficients are quantized by all possible $QP$ values to get the $(\rho, QP)$ table; then, all possible $QP$ values are checked to select the proper $QP$ for a given $\rho$. For fair comparison, the proposed rate control scheme and the original $\rho$ domain rate control both adopt the frame level bit allocation and initialization of $QP$ for RDO presented in Section IV-C. In addition, it is worthwhile to note that the MB level bit allocation can not be integrated into the original $\rho$ domain rate control (see Section II), since the $QP$ for the current MB is obtained using the target fraction of zero coefficients for the remaining MBs.

### A. Video quality in PSNR

Table II list the PSNR and bit rate (BR) of the proposed rate control ("$Proposed$") and the original $\rho$ domain rate control ("$Original$"). From Table II, it can be seen that the proposed method can achieve better PSNR than the original method for most tested sequences. This is because the proposed method adopts the MB level bit allocation, which can improve the frame quality by properly distributing the bits among all MBs. One can also see that the proposed method has worse PSNR than the original method on $Football$ at low bit rate. This is because the spatial and temporal content of $Football$ are very complex. The effect of MB level bit allocation is reduced at low bit rate due to the limited target bit budget.

### B. Bit accuracy of rate control

From Table II, one can also see that the actual number of bits produced by the proposed method is much closer to the target bit rate when compared to the original method. To further compare the bit accuracy, Fig. 3 and 4 illustrate the actual bits produced for each frame for the $Football$

TABLE II
PERFORCE COMPARISON BETWEEN THE PROPOSED ALGORITHM AND THE ORIGINAL ONE IN TERMS OF BIT RATE AND PSNR

| Sequences | Target BR (kbps) | Original | | Proposed | | |
|---|---|---|---|---|---|---|
| | | BR (kbps) | PSNR (dB) | BR (kbps) | PSNR (dB) | PSNR Gain (dB) |
| Bus | 300 | 242.86 | 25.29 | 293.40 | **25.75** | 0.46 |
| | 500 | 501.81 | 27.28 | 490.22 | **27.45** | 0.17 |
| | 1000 | 995.57 | 28.90 | 985.75 | **29.60** | 0.70 |
| | 2000 | 1998.32 | 30.12 | 1981.65 | **33.41** | 3.29 |
| Container | 300 | 237.09 | 34.11 | 292.32 | **34.30** | 0.19 |
| | 500 | 441.88 | 35.68 | 494.86 | **36.50** | 0.82 |
| | 1000 | 946.73 | 37.21 | 990.90 | **39.01** | 1.80 |
| | 2000 | 1962.52 | 38.10 | 1997.27 | **40.78** | 2.68 |
| Football | 300 | 228.21 | **26.51** | 295.42 | 26.35 | -0.16 |
| | 500 | 450.80 | **29.11** | 493.61 | 29.03 | -0.08 |
| | 1000 | 962.33 | 32.38 | 989.38 | **32.78** | 0.40 |
| | 2000 | 1968.35 | 35.80 | 1977.37 | **37.12** | 1.32 |
| Foreman | 300 | 216.62 | 30.82 | 306.21 | **32.00** | 1.18 |
| | 500 | 443.36 | 33.74 | 508.10 | **34.21** | 0.47 |
| | 1000 | 952.23 | 35.76 | 1004.56 | **36.70** | 0.94 |
| | 2000 | 1957.95 | 37.13 | 2000.21 | **38.99** | 1.86 |
| Mobile | 300 | 251.08 | 24.07 | 298.95 | **24.47** | 0.40 |
| | 500 | 460.68 | 25.63 | 497.75 | **26.66** | 1.03 |
| | 1000 | 968.48 | 27.43 | 995.47 | **29.81** | 2.38 |
| | 2000 | 1974.51 | 29.13 | 1990.37 | **33.37** | 4.24 |

and $Foreman$ sequences at 300 $kbps$. We can find that the fluctuation of the actual bits produced by each frame in the proposed method is much smaller for the original method. This improvement is due to the switched $QP$ calculation scheme in the proposed method.

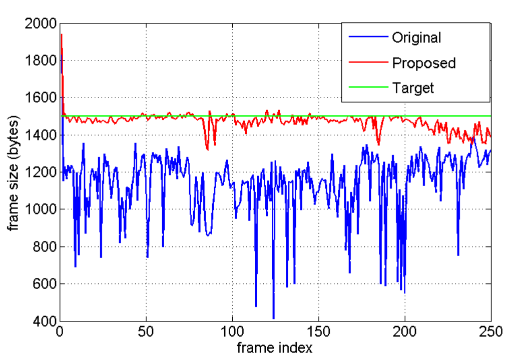


Fig. 3. Bit rate for individual frames for $Football$ at 300 kbps.

Table III presents the average deviation of the frame size from the target bit budget for each sequence, which is calculated with (13).

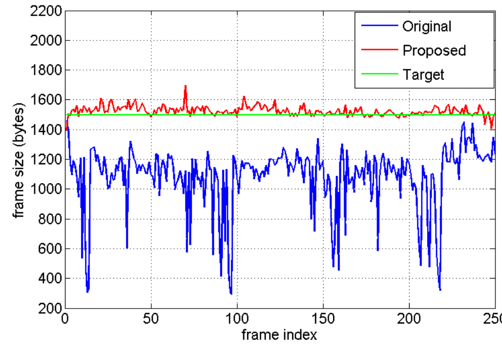


Fig. 4. Bit rate for individual frames for $Foreman$ at 300 kbps.

$$Dev = \frac{1}{T} \cdot \sum_{j} \frac{|R^j_{actual} - R_T|}{R_T} \cdot 100\% \qquad (13)$$

where $R^j_{actual}$ is the frame size produced by frame $j$, $R_T$ is the target bit budget of each frame, and $T$ is the total number of encoded frames in a sequence.

TABLE III
AVERAGE DEVIATION OF THE FRAME SIZE FROM THE TARGET BIT BUDGET AND MAXIMUM FRAME SIZE

| Sequences | Target frame size $R_T$ (byte) | Original Dev[%] | Proposed Dev[%] |
|---|---|---|---|
| Bus | 1500 | 28.05 | **2.94** |
| | 2500 | 7.18 | **2.55** |
| | 5000 | 2.12 | **1.45** |
| | 10000 | 0.81 | **0.65** |
| Container | 1500 | 21.61 | **3.63** |
| | 2500 | 11.93 | **1.72** |
| | 5000 | 5.47 | **1.08** |
| | 10000 | 1.95 | **0.60** |
| Football | 1500 | 24.62 | **2.30** |
| | 2500 | 10.26 | **1.82** |
| | 5000 | 3.92 | **1.21** |
| | 10000 | 1.67 | **1.06** |
| Foreman | 1500 | 28.37 | **1.94** |
| | 2500 | 11.77 | **1.53** |
| | 5000 | 4.94 | **0.76** |
| | 10000 | 2.17 | **0.42** |
| Mobile | 1500 | 17.56 | **1.61** |
| | 2500 | 8.60 | **1.41** |
| | 5000 | 3.39 | **1.17** |
| | 10000 | 1.37 | **0.49** |

From Table III, it can be seen that the proposed method has the smallest deviation, which indicates that it can control the frame size more accurately and make full use of the transmission capacity of the channel.

### C. Computational complexity

In the original $\rho$ domain rate control, the transform coefficients are quantized by all possible $QP$ values to get the $(\rho, QP)$ table. For a given $\rho$, it checks all possible $QP$ values to select the proper $QP$. In the proposed method, the transform coefficients are only quantized with two $QP$s to calculate the model parameters in (3). For a given $\rho$, the $QP$ can be calculated with the method in Subsection IV-B. Thus, the computational complexity of the proposed method is much lower than the original one. Here, we use the reduction of the encoding time to measure the computational complexity of the two methods, which is defined as follows:

$$\Delta_C = \frac{C_{Org} - C_{Pro}}{C_{Org}} \cdot 100\% \quad (14)$$

where $C_{Org}$ and $C_{Pro}$ are the encoding time of the original method and the proposed method, respectively.

TABLE IV
THE ENCODING TIME REDUCTION OF THE PROPOSED METHOD RELATIVE TO THE ORIGINAL METHOD

| Sequences | Target BR (kbps) | Encoding time reduction [%] |
|---|---|---|
| Bus | 300 | 52.79 |
| | 500 | 48.64 |
| | 1000 | 46.06 |
| | 2000 | 45.13 |
| Container | 300 | 45.90 |
| | 500 | 45.72 |
| | 1000 | 42.79 |
| | 2000 | 44.39 |
| Football | 300 | 58.17 |
| | 500 | 52.87 |
| | 1000 | 50.29 |
| | 2000 | 45.44 |
| Foreman | 300 | 49.82 |
| | 500 | 49.03 |
| | 1000 | 42.50 |
| | 2000 | 40.28 |
| Mobile | 300 | 56.67 |
| | 500 | 56.00 |
| | 1000 | 52.22 |
| | 2000 | 48.40 |

The reduction of encoding time for each sequence is shown in Table IV. From Table IV, it can be seen that the reduction of the encoding time is between 40% and 58%. Thus, the proposed method is more suitable for low delay video communication.

## VI. CONCLUSION

In this paper, we proposed a MB level rate control algorithm for low-delay video communication based on the $\rho$ domain rate model. In the proposed rate control scheme, an exponential model is used to describe the relationship between $\rho$ and $Qstep$, with which the $QP$ for each MB can be obtained in an efficient way. Furthermore, a switched $QP$ calculation scheme is proposed to avoid large deviations of the actual frame size from the target bit budget. Compared with the original $\rho$ domain rate control, the proposed method can achieve better video quality, and higher bit rate accuracy. Meanwhile, the encoding time is also significantly reduced.

## VII. ACKNOWLEDGEMENT

This work was supported in part by the Major State Basic Research Development Program of China (973 Program 2015CB351804) and the National Natural Science Foundation of China under Grant No. 61272386 and in part by, the European Research Council under the European Union's Seventh Framework Programme (FP7/2007-2013)/ERC Grant agreement no. 258941.

## REFERENCES

[1] Z. G. Li, W. Gao, F. Pan, S. W. Ma, K. P. Lim, G. N. Feng, X. Lin, S. Rahardja, H. Q. Lu, and Y. Lu, "Adaptive rate control for H.264", J. Visual Comm. Image Representation, vol. 17, no. 2, pp. 376-406, Apr. 2006.

[2] M. Jiang and N. Ling, "Low-delay rate control for real-time H.264/AVC video coding," IEEE Trans. Multimedia, vol. 8, no. 3, pp. 467-477, Jun. 2006.

[3] J. Dong and N. Ling, "A context-adaptive prediction scheme for parameter estimation in H.264/AVC macro-block layer rate control", IEEE Trans. Circuits Syst. Video Technol., vol. 19, pp. 1108-1117, Aug. 2009.

[4] Z. He and S. K. Mitra, "A linear source model and a unified rate control algorithm for DCT video coding," IEEE Trans. Circuits Syst. Video Technol., vol. 12, pp. 970-982, Nov. 2002.

[5] Z. He, Y.K. Kim, S.K. Mitra, "Low-delay rate control for DCT video coding via domain source modeling," IEEE Trans. Circuit and System for Video Technology, vol.11, no.8, pp.928-940, Aug. 2001.

[6] Z. He and S. K. Mita, "Optimum bit allocation and accurate rate control for video coding via domain source modeling," IEEE Trans. on Circuits and Systems for Video Technology, vol. 12, no. 10, pp. 840-849, Oct. 2002.

[7] Z. He and D. O. Wu, "Linear rate control and optimum statistical multiplexing for H.264 video broadcast," IEEE Trans. Multimedia, vol. 10, no. 7, pp. 1237-1249, Nov. 2008.

[8] F. Zhang and E. Steinbach, "Improved domain rate control with accurate header size estimation," in Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP), Prague, May. 2011, pp. 813-816.

[9] D. Kwon, M.Y. Shen, and C.J. Kuo, "Rate control for H.264 video with enhanced rate and distortion models," IEEE Trans. Circuits Syst. Video Technol., vol. 17, pp.517-529, May. 2007.

[10] M. Liu, H. Q. Li and C. W. Chen, "Low-complexity rate control based on domain model for scalable video coding," in Proc. IEEE Int. Conf. Image Processing, Hong Kong, Sept. 2010, pp. 1277-1280.