Adaptive Intra-Refresh for Low-Delay Error-Resilient Video Coding

Haoming Chen[*], Chen Zhao[**], Ming-Ting Sun[*], and Aaron Drake[†]

[*] University of Washington, Seattle, WA, USA

[**] Peking University, Beijing, China

[†] T-Mobile USA, Bellevue, WA, USA

E-mail: [*] eehmchen@uw.edu, [**] zhaochen@pku.edu.cn, [*] sun@ee.washington.edu, [†] Aaron.Drake@T-Mobile.com

***Abstract*— Low-delay and error-resilience video coding are critical for real-time video communication over wireless networks. Intra-refresh coding, which embeds intra-coded regions into inter frames can achieve a relatively smooth bit-rate and terminate the error propagation caused by the transmission loss. In this paper, we proposed a novel linear model for the intra-refresh cycle size selection adapting to the network packet loss rate and the motions in the video content. Experimental results show that this linear model works efficiently. The modelled cycle size can achieve almost the same quality as the optimal cycle size under different packet loss rates.**

***Index Terms*— Error-resilience, intra-refresh coding, low delay, video coding, visual communication**

I. Introduction

Low-delay and error-resilience are critical for real-time video chat applications over wireless networks. On the delay aspect, given a fixed network throughput, the delay induced by the video codec mostly depends on the video encoder buffer size. On the error-resilience aspect, the error induced by the packet loss will propagate to the following several frames due to the nature of the predictive coding framework. To terminate an error-propagation, conventional video coding schemes encode an intra-coded frame (I-frame) in each Group-Of-Pictures (GOP). An I-frame is encoded without any reference to other frames, and so, it is not affected by errors in the previous frames. For a GOP with a size of K frames, in the worst case when the beginning I-frame is lost, error-propagation is limited to within the following K-1 Predictive frames (P-frames) assuming for low-delay applications the B-frames (Bi-directional predicted frames) are not used. However, for the low delay aspect, the GOP coding structure may not be the best choice. Since an I-frame only exploits the spatial redundancy within itself, it generates much more bits than a P-frame. The resulting bitstream usually needs to be smoothed by a large encoder buffer to ensure it does not exceed the network transport capability. This rate-smoothing encoder buffer could cause relatively long delay.

Intra-refresh (or intra-slice) coding schemes can provide low-delay and good error-resilience features. In intra-refresh coding, instead of encoding the whole frame as an I-frame, a subset of Macro Blocks (MBs) in each frame can be forced into intra-coded MBs, so that after a number of frames, the whole frame is completely refreshed. This spreading of intra MBs into the P-frames can produce a relatively uniform bit-rate. With the uniform bit-rate, the encoder buffer could be avoided or kept to a minimal size to result in a low-delay video codec. Besides offering low-delay, intra-refresh coding schemes also provide good error-resilience performance. Since the whole frame is completely refreshed after a cycle of frames, the parts of the picture affected by the transmission errors will be constantly refreshed. A vertical partition intra-refresh scheme as shown in Fig. 1 is applied in x264 [1], a popular open-source software, encoding videos into the H.264/MPEG-4 AVC format. In the vertical partition intra-refresh scheme, given the intra-refresh cycle size N, the whole frame is split into N regions vertically. In the mode decision process, the MBs within each intra-refresh region (shaded area in Fig. 1) are forced to be intra-coded.

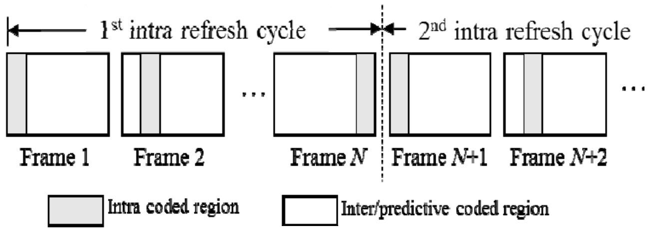

Fig. 1. Vertical partition of intra-refresh regions.

To achieve a better performance under different network conditions, a fixed-cycle intra-refresh scheme is not optimal. In [2], a rate-distortion model is proposed considering the channel (network packet loss rates) and the source (the intra coded MB percentage) jointly. Based on this model, given the network condition, the optimal number of intra coded MB can be derived. However, some empirical and sequence-dependent parameters make the derivation of optimal intra coded MB percentage (and thus the number of frames in an intra-refresh cycle) difficult. In [3], a motion-adaptive intra-refresh order is proposed. However, only a fixed-cycle intra-refresh scheme is studied in that paper.

In this paper, a content-based linear cycle size selection model is proposed. Given the packet loss rate and the motion information, the best intra-refresh cycle size is estimated. Compared with [3], our model is much simpler and has no empirical and sequence-dependent parameter. Experimental results show that the cycle size selection model is very effective.

The rest of this paper is organized as follows. In Section II, a linear model is proposed for the adaptive selection of the cycle size. In Section III, the model parameters are determined. Section IV shows the simulation results. Section V concludes this paper and describe possible future work.

II. Linear Model for Adaptive Cycle Size Selection

Fig. 2 compares different refresh cycle sizes N (N = 4, 8, 16 and 32) under different packet loss rates (10^{-4} ~ 10^{-1}). A smaller cycle is better for recovering the error more quickly when the packet loss-rate is high, and a large refresh cycle is better for a low packet-loss rate since it gives better coding gains with a smaller intra-coded area in each frame. Thus, for optimal performance, the refresh cycle size N should be adaptive to the packet loss rates. In this section, we propose a linear model to predict the optimal intra-refresh rate based on the network packet loss rate and the motion in the video content.

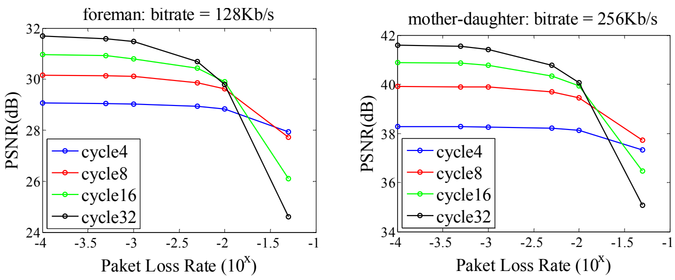

Figure 2. Simulation results for the test sequences "foreman" and "mother-daughter" encoded by x264 with the vertical partition intra-refresh scheme [2]. The sequences are encoded with four different cycle sizes (4, 8, 16 and 32). Each random packet loss causes a frame loss. The loss frame is concealed by repeating the previous frame.

In [3], a joint end-to-end distortion model is proposed w.r.t. the intra-refresh rate and the packet loss rate:

$$D_s(R_s,\beta) = D_s(R_s,0) + \beta(1-\lambda+\lambda\beta)\times[D_s(R_s,1) - D_s(R_s,0)] \quad (1)$$

$$D_c = \frac{a}{(1-b+b\beta)}\frac{p}{1-p}E[F_d(n,n-1)] \quad (2)$$

In Eq. (1), D_s is the source distortion, which is a function of the source bitrate R_s and the intra-refresh rate β (= number of intra coded MBs/total number of MBs) in one frame. λ is a sequence-dependent parameter. The source distortion is a linear combination of two extreme cases: all MBs inter-coded $D_s(R_s,0)$ and all MBs intra-coded $D_s(R_s,1)$. In Eq. (2), D_c is the channel distortion caused by the packet loss rate p. Parameter a is a constant, and b is a constant describing the motion randomness of the video scene. $E[F_d(n,n\text{-}1)]$ is the expectation of the difference of the neighbouring frames n and n-1.

The total end-to-end distortion is the sum of the source distortion and the channel distortion:

$$D = D_s + D_c. \quad (3)$$

Under a bitrate R_s and the packet loss rate p, the best intra-refresh rate that minimizes the end-to-end distortion can be derived by taking the derivative of D w.r.t. β and setting it to zero, resulting in a cubic equation of β,

$$A\beta^3 + B\beta^2 + C\beta + D = H\frac{p}{1-p} \quad (4)$$

where

$$A = 2\lambda b^2, \quad (5)$$

$$B = b(b+4\lambda-5\lambda b), \quad (6)$$

$$C = 2(1-b)(b+\lambda-2\lambda b), \quad (7)$$

$$D = (1-b)^2(1-\lambda), \text{ and} \quad (8)$$

$$H = \frac{abE[F_d(n,n-1)]}{D_s(R_s,1) - D_s(R_s,0)}. \quad (9)$$

We observe that the intra-refresh rate β is the inverse of cycle size N, so β is always much smaller than 1, e.g., N = 12 means β = 0.083. Also, the coefficients A, B, C, and D in Eqs. (5-9) are constants. Hence, we propose to omit the higher order items and simplify Eq. (4) into a linear model, as follow:

$$\beta = H'\frac{p}{1-p} + D' \quad (10)$$

where $H' = H/C$ and $D' = -D/C$, which are constants (dependent on the sequences). From Eq. (10) the optimal refresh rate is linear to $p/(1\text{-}p)$.

To verify that the proposed linear model is sufficiently accurate, we test 21 video sequences as shown in Fig. 3. Each sequence is encoded at the 1Mbps bitrate with the cycle size N from 4 to 40. Note that since the model is independent to the intra-refresh pattern (random intra-refresh and vertical partition intra-refresh [2]), we use the random intra-refresh pattern. We simulate the packet loss rate from 0.1% to 20%. Each packet loss scenario is simulated 50 times with randomly selected lost packets. For each packet loss rate, the average PSNR of decoded video is calculated compared with original video and we select the size of frames in a cycle (N) that maximizes the average PSNR and plot β (i.e., $1/N$) versus $p/(1\text{-}p)$ in Fig. 3. From the testing results and the fitting line, we can see that the linear model works well.

III. Parameters Training of the Linear Model

With linear model in Eq. (10), given the parameters H' and D' we can estimate the best cycle size easily. However, since C, D, and H depend on many empirical constants, it is difficult to get a closed-form solution of H' and D'. In this section, we propose to approximate these parameters by training using some video sequences.

We plot the 21 corresponding fitting lines in Fig. 4. We have following observations on the slope of these lines: there is a rough relationship between the slope of the linear model and the motion of video content. For example, "ice" and "highway" sequences have very fast motion and they have the largest slopes; “container” and “bridge-close” have very slow motion and have the smallest slopes.

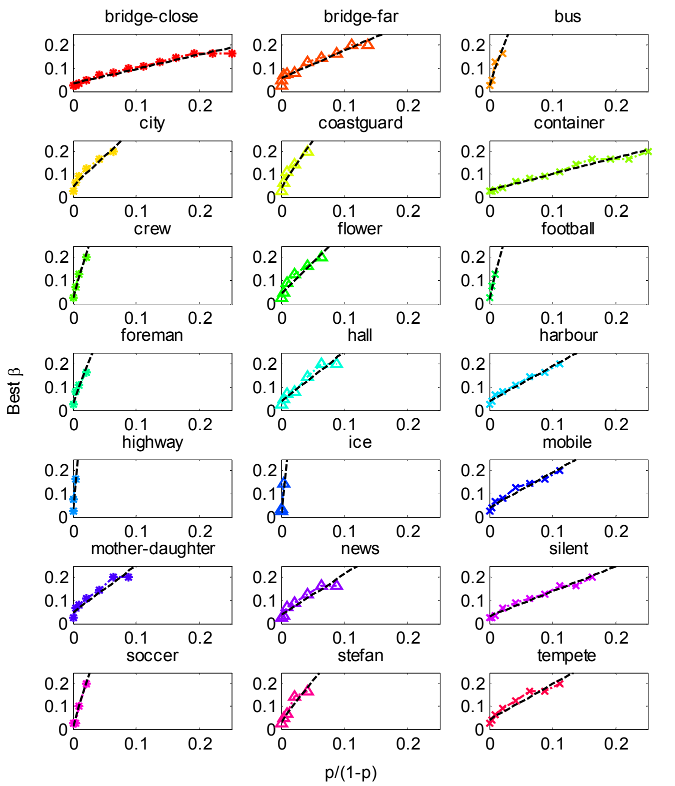

Figure 3. Testing results on 21 sequences. The x-axis is p/(1-p), which is in the range of (0.01, 0.25] and the y-axis is the best β, i.e., 1/(Best Cycle Size), which is in the range of [1/40, 1/4]. Here the “Best Cycle Size” is the cycle size (between 4 and 40) that minimizes the distortion for each packet loss rate. Note that for some sequences, the best cycle size reaches the smallest cycle size limited to 4 ($\beta=0.25$), and so the line becomes flat. We have removed that flat part to show the linearity clearly. The dashed line is the fitting linear function. We can see that the linear model can approximate the relationship between β and $p/(1\text{-}p)$ well.

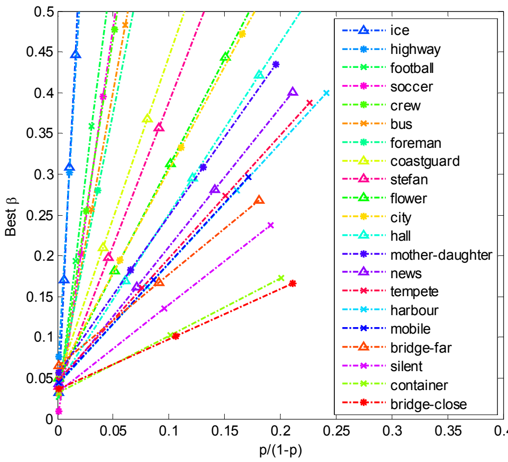

Figure 4. Fitting lines of the 21 sequences in the same plot. (Best viewed in colour.)

Actually, these observations can be justified from the expressions of C, D and H in Eqs. (7-9). For example, H', which equals to H/C is directly proportion to $E[F_d(n,n\text{-}1)]$ and inversely proportional to $D_s(R_s,1) - D_s(R_s,0)$. $E[F_d(n,n\text{-}1)]$ is the difference of neighboring frames, which is large when the motion of the video content is fast. Moreover, for a fast motion video, the error propagates quickly, so a large intra-refresh rate (or a small cycle size) is needed. For $D_s(R_s,1) - D_s(R_s,0)$, the distortion difference between all-intra and all-inter under the same bitrate, if it is large, it means force-intra coded MBs will induce more distortion, so a small intra-refresh rate is preferred, resulting in a smaller slope in the linear model.

Based on the above analysis, we model the slope proportional to $E[F_d(n,n\text{-}1)]/[\,D_s(R_s,1) - D_s(R_s,0)]$. To see the proportion, in Fig. 5, we plot the slope value H' versus $E[F_d(n,n\text{-}1)]/[D_s(R_s,1) - D_s(R_s,0)]$. For each sequence, $E[F_d(n,n\text{-}1)]$ is the average MSE of all neighbouring frame pairs and the $D_s(R_s,1) - D_s(R_s,0)$ is the average MSE difference between two cases: all intra coding and all inter coding structures. Then, we fit the points with a linear model, which results in the following model:

$$H' = c \times E[F_d(n,n\text{-}1)]/[\,D_s(R_s,1) - D_s(R_s,0)] + d. \qquad (11)$$

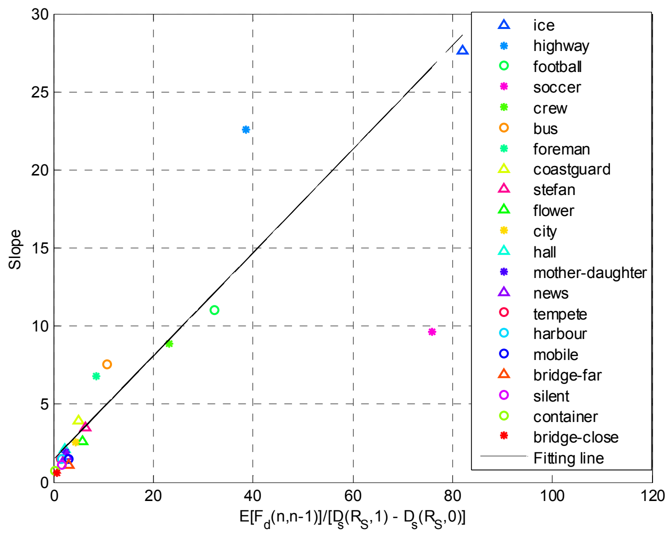

Figure 5. The slopes of the linear models versus the proportion of $E[F_d(n,n\text{-}1)]$ and $D_s(R_s,1) - D_s(R_s,0)$. (Best viewed in colour.)

To verify the trained model, we separate these 21 sequences into two sets: 11 sequences for training the parameters in Eq. (11) and the other 10 sequences for testing the performance of our proposed cycle size selection model, the testing results are shown in Section IV. In the training part, the parameters c = 0.3164 and d = 1.6625.

For the offset points (D' in Eq. (10)) of each line in Fig. 4, they have very close positions, so, in the linear model, we just use the average value of the offsets in the training set as the offset of the proposed model.

Combining Eq. (10) and Eq. (11), a selection model of cycle size N is as follow:

$$\beta = \frac{1}{N} = \left[\frac{0.3164 \times E[F_d(n,n-1)]}{D_s(R_s,1) - D_s(R_s,0)} + 1.6625\right] \times \frac{p}{1-p} + 0.0342 \qquad (12)$$

In low-delay video coding, we cannot get the neighboring frame difference $E[F_d(n,n\text{-}1)]$, the all-intra MSE $D_s(R_s,1)$ and all-inter MSE $D_s(R_s,0)$ of the whole sequence. One method is to estimate these values based on the first N cycle. However, this may not work when the video content changes. Another method, which is more accurate, is to re-calculate these values every several frames and update them adaptively. Since the test sequences do not have scene changes, we use the former method.

IV. Experimental Results

In this test, we use Eq. (12) to determine the cycle size and apply it on the testing set of 10 sequences. Some testing condition are listed in Table I.

TABLE I
TEST CONDITIONS

Reference software	JM 18.6 [4]
Profile	Main
RD optimization	Yes
Rate control	1 Mbit/s
Number of reference frames	1
Intra-refresh scheme	Random intra-refresh

We compare following four different cases and the results are shown in Fig. 6.

1) Optimal cycle size that maximizes the PSNR of decoded video.
2) The cycle size is determined by our proposed selection model Eq. (12).
3) Cycle size is fixed as 20.
4) Cycle size is fixed as 10.

We have following remarks for these results:

1) Our proposed linear model for cycle selection can approximate the optimal cycle size very well. We also show the PSNR difference in Table II. The quality difference compared to the optimal cycle size is very small.
2) Compared to intra-refresh with a fixed cycle size N = 20 and 10, more than 1 dB PSNR difference can be seen especially at high packet loss rate.
3) In this test, only relatively low resolution sequences are tested. More tests would be applied on HD sequences to verify the model on higher resolutions.

TABLE II
PSNR (IN DB) DIFFERENCE BETWEEN OPTIMAL CYCLE SIZE AND MODELLED CYCLE SIZE

Sequences	p = 0.1%	p = 1%	p = 10%	p = 20%
harbour	-0.09	-0.03	-0.03	0.00
highway	-0.03	-0.03	0.00	0.00
ice	-0.15	0.00	0.00	0.00
mobile	-0.23	0.00	-0.09	0.00
mother-daughter	-0.09	-0.02	0.00	0.00
news	-0.33	-0.04	0.00	0.00
silent	-0.29	-0.15	-0.31	0.00
soccer	-0.37	-0.22	0.00	0.00
stefan	-0.25	0.00	0.00	0.00
tempete	-0.09	-0.01	-0.06	0.00
Average	-0.19	-0.05	-0.05	0.00

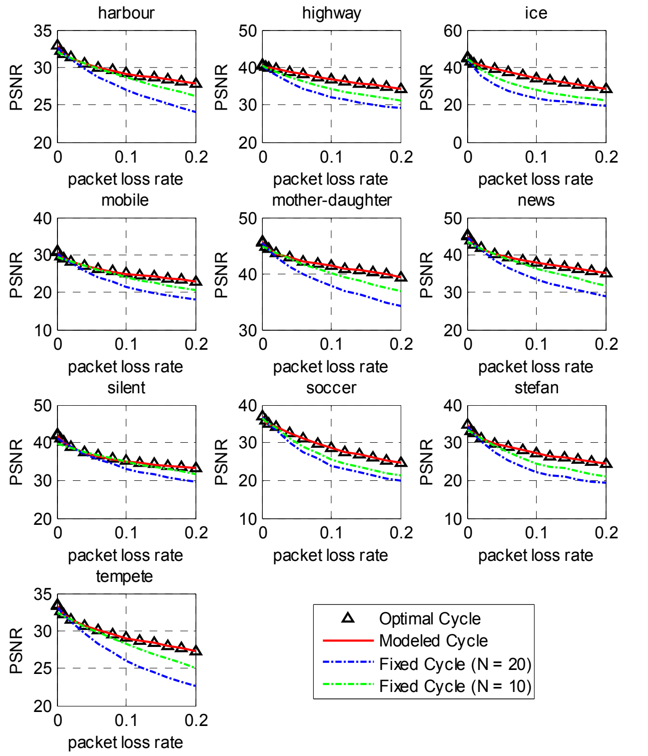

Figure 6. Comparison of the proposed cycle size selection algorithm, optimal cycle size, and fixed cycle size (10 and 20). The PSNR is for the decoded video sequences. The lost frame is concealed with copying the previous decoded frame.

V. Conclusions and Future Work

In this paper, we present an efficient intra-refresh cycle size selection model depending on the network packet loss rate. To select the best cycle size adaptively according to the packet loss rate, we propose a linear model, where no heuristic sequence-specific parameters are needed. Experimental results confirm the effectiveness of the proposed algorithm.

In current simulation, one intra-refresh cycle size is used for the whole video sequence, which does not consider the scene change in the video. One of our future work is to extend this scheme so that the parameters can be updated more frequently and the cycle size selection can be adaptive to the scene changes. Another future work is to extend the proposed approach to the HEVC standard.

References

[1] "VideoLAN–x264, the best H.264/AVC encoder," http://www.videolan.org/developers/x264.html.

[2] Z. He, Cai J., and C.W. Chen. "Joint source channel rate-distortion analysis for adaptive mode selection and rate control in wireless video coding." *IEEE Transactions on Circuits and Systems for Video Technology*, 12.6 (2002): 511-523.

[3] R.M. Schreier and A. Rothermel, "Motion adaptive intra refresh for the h.264 video coding standard," *IEEE Transactions on Consumer Electronics*, vol.52, no.1, February 2006, pp.249-253.

[4] H.264/AVC Reference Software, available at http://iphome.hhi.de/suehring/tml/download/jm18.6.zip, [Online; accessed 6-Feb-2014]