|  |  |
| --- | --- |
| **Joint Collaborative Team on Video Coding (JCT-VC)**  **of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11**  7th Meeting: Geneva, CH, 21-30 November, 2011 | Document: JCTVC-G278 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | **CE4 Subtest 2: Adaptive De-Quantization Offset** | | |
| *Status:* | Input Document to JCT-VC | | |
| *Purpose:* | Proposal | | |
| *Author(s) or Contact(s):* | \*Xiang Li  \*Xun Guo  #Shawmin Lei  \*North Building 10F, Raycom Infotech Park Tower C, No. 2 Kexueyuan South Rd., Haidian District, Beijing, China 100190  #No. 1, Dusing Rd. 1, Hsinchu Science Park, Hsinchu, Taiwan 30078 | Email: | [x.li@mediatek.com](mailto:x.li@mediatek.com)  [xun.guo@mediatek.com](mailto:xun.guo@mediatek.com)  [shawmin.lei@mediatek.com](mailto:shawmin.lei@mediatek.com) | |
| *Source:* | MediaTek Inc. | | |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Abstract

This contribution presents experimental results for the proposal on adaptive de-quantization offset (AQO) in Core Experiment 4 Subtest 2. Specifically, a slice level AQO is investigated and proposed. It is observed that on average 1.3% luma PCHIP-BD-rate reduction and around 2% chroma PCHIP-BD-rate reduction were obtained for four inter configurations (RA-HE, RA-LC, LB-HE, and LB-LC) defined in JCTVC-F900. It is also observed that the average RVM values of the proposed method are quite similar to those of HM-4.0 anchor. Moreover, it is reported that in low QP scenarios (QP=2, 7, 12, 17), 0.4%, 1.6% and 2.3% PCHIP-BD-rate reductions were obtained on average for the four inter configurations.

# Introduction

Adaptive de-quantization offset (AQO) was initially proposed in JCTVC-E091 [1] and then in JCTVC-F119 [2]. The general idea of AQO is to adaptively select de-quantization offset based on input video content.

Compared to JCTVC-F119, the following parts are improved or modified in this contribution:

* Removed lambda refinement (encoder only)
* Revised statistics collection and de-quantization offset calculation (encoder only)
* Harmonized with QP rounding [3] (encoder only)
* Harmonized with EEM [4] (encoder only)
* New syntax elements to reduce the bit cost of AQO signaling (optional, enabled by default)

The second part is the same as that proposed in JCTVC-F276 [5]. The last part is described in section A.3. Other parts and related implementation details are discussed in section 2.

# Algorithm Description

## General idea of de-quantization offset

In current HEVC and HM-4.0 software, de-quantization process is performed as follows.



where  and  are reconstructed and quantized coefficients, respectively,  is related quantization step.

To further improve coding efficiency, [1][2] proposed to include a de-quantization offset in , namely



where  is the so-called de-quantization offset. Generally,  is adaptively selected according to coding statistics of transformed coefficients.

## De-quantization offset determination at frame level

During slice initialization at encoder, the de-quantization offsets are derived for Y, U and V components according to the statistics of transformed coefficients in preceding same-type-same-temporal-level frames.

Let ,  and  denote the original non-zero transformed coefficient, the quantized non-zero transformed coefficient and the related quantization step in the preceding same-type-same-temporal-level frame, the de-quantization offset  of the current frame is calculated as



where *N* indicates the number of quantized non-zero transformed coefficients in the preceding same-type-same-temporal-level frame. When the statistics of preceding same-type-same-temporal-level frame is not available such as the first frame,  is set to zero.

After the calculation with ,  further is clipped to non-negative value unless for intra frames.

## Harmonization with QP rounding

QP rounding is an efficient encoder optimization method, which is based on the fine-QP method proposed in [6]. Essentially, fine-QP adaptively selects fractional QP for the current frame according to the statistics of transformed coefficients. QP rounding rounds the fractional QP to the closest integer QP and uses the integer QP for the current frame.

To harmonize with QP rounding, the proposed de-quantization offset is determined after QP update by QP rounding. Moreover, when the QP of one frame is changed by QP rounding, the proposed de-quantization offset is set to zero since the statistics used to determine the de-quantization offset is based on unchanged QP.

## Harmonization with EEM

EEM [4] is an efficient encoder optimization method, which adaptively determines rounding offset *r* in the quantization process formulated in .



Basically, AQO and EEM are orthogonal since they are applied in quantization and de-quantization respectively. To harmonize with EEM, the de-quantization offset needs to be considered when determining quantized coefficient  with . It is proposed to use instead of for the quantization process.



# Simulations and Discussions

To verify the performance of the proposed algorithm, four tests (test A – test D) defined in JCTVC-F904 [3] were conducted. Detailed results can be found in attached excel files. Note that QP rounding is not enabled in test cases other than test C. When enabled QP rounding, higher coding gain can be obtained by AQO. Please also note that the current timing information is not reliable. Basically, AQO does not change encoding and decoding time much, as verified in [1][2].

## Test A

In this test case, AQO is compared to HM4.0 anchor.

On average, 1.3%, 1.9% and 2.0% PCHICP-BD-rate savings were obtained for Y, U and V components, respectively. The average RVM value of AQO is quite similar to HM4.0 anchor.

Table Test A



## Test B

In this test case, AQO is compared to HM4.0 anchor with following rate constraint: The average bitrate mismatch between AQO and HM4.0 anchor shall be within 2% for each temporal level of in the tests of all sequences and QP settings. For the purpose of obtaining the target bitrate of each temporal level, Lambda tuning is allowed.

On average, 1.0%, 0.6% and 0.7% PCHIP-BD-rate savings were obtained for Y, U and V components, respectively.

To meet the rate constraints, a script tool [10] was used to iteratively tune lambda. Note that this tool cannot guarantee a good RD performance. Therefore, the coding performance is not as good as that reported in test A.

Table Test B



## Test C

In this test case, AQO is compared to HM4.0 + QP rounding [3]. On average, 0.3%, 2.8% and 2.9% PCHIP-BD-rate savings were obtained for Y, U and V components, respectively.

Table Test C



## Test D

In this test case, AQO is compared to HM4.0 with RDOQ off and EEM [4] on.

On average, 1.2%, 1.3% and 1.6% PCHIP-BD-rate savings were obtained for Y, U and V components, respectively, which indicates that AQO is almost orthogonal to EEM.

Table Test D



## Additional results

In this section, additional results are provided. Please note that results in section 3.5.2 and 3.5.3 were obtained with updated software.

### Test A with syntax in section A.3 disabled

To verify the performance of newly added syntax on signaling cost reduction, test A is performed while disabling the syntax in section A.3. Compared to the test A results in section 3.1, it can be observed that 0.1% luma PCHIP-BD-rate is saved by the proposed syntax.

Table Test A with syntax in A.3 disabled



### Test A with QP rounding enabled (updated software)

When enabled QP rounding, a better coding performance can be obtained by AQO. On average, 1.4%, 6.5% and 6.9% PCHICP-BD-rate savings were obtained for Y, U and V components, respectively.

Table Test A with AQO and QP rounding both enabled



### Test A with low QP settings

As suggested in JCTVC-G403 [9], test A with low QP settings (QP=2, 7, 12, 17) is performed in this section. Please note that the reference data is copied from JCTVC-G403.

On average, 0.4%, 1.6%, 2.3% PCHIP-BD-rate reductions were obtained for Y, U, and V components, respectively.

Table Test A with low QP settings (QP=2, 7, 12, 17)



# Conclusions

In this contribution, adaptive de-quantization offset (AQO) is proposed. Compared to the four inter cases of HM4.0 anchor, average PCHIP-BD-rate reduction of 1.3%, 1.9%, and 2.0% is obtained for Y, U, and V components, respectively. When enabled QP rounding (an encoder only optimization method), higher coding gain can be obtained. Moreover, the computational complexity of AQO is quite low. Considering the gain by AQO is stable in all tested cases including low QP cases, it is proposed to adopt AQO into HEVC and reference software.

# References

1. X. Li, X. Guo, and S Lei, “Adaptive De-quantization Offset”, JCTVC-E091, Geneva, Switzerland, Mar. 2011.
2. X. Li, X. Guo, and S Lei, “CE4 Subtest3: Adaptive De-Quantization Offset”, JCTVC-F119, Torino, Italy, Jul. 2011.
3. K. Sato, M. Budagavi, M. Coban, H. Aoki and X. Li, “Description of core experiment 4: Quantization”, JCTVC-F904, Torino, Italy, Jul. 2011.
4. Gary Sullivan, “Adaptive quantization encoding technique using an equal expected-value rule”, JVT-N011, Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG, Hong Kong, China 18-21 Jan., 2005
5. X. Yu, J. Wang, D. He, E. Yang, “CE4-subtest 3.3.2: Quantization with Adaptive Reconstruction Levels”, JCTVC-F276, Torino, Italy, Jul. 2011.
6. X. Wang, R. Joshi, G. Van Der Auwera, M. Karczewicz, “Fine granularity QP change at slice level”, JCTVC-F276, Torino, Italy, Jul. 2011.
7. Frank Bossen, “Common test conditions and software reference configurations”, JCTVC-F900, Torino, Italy, Jul. 2011.
8. B. Bross, W.-J. Han, J.-R. Ohm, G. J. Sullivan, T. Wiegand, “WD4: Working Draft 4 of High-Efficiency Video Coding”, JCTVC-F803, Torino, Italy, Jul. 2011.
9. B. Li, J. Xu, G. Sullivan, “CE4-Subtest-2 Cross-check of Adaptive Reconstruction Levels for Quantization Design (JCTVC-G382)”, JCTVC-G403, Geneva, Switzerland, Nov. 2011.
10. S. Campbell, J. Wang, X. Yu, “A tool for rate-constrained performance test”, JCTVC-G678, Geneva, Switzerland, Nov. 2011.

# Patent rights declaration(s)

NOTE – Activities in the JCT-VC and contributions to the JCT-VC are subject to the common patent policy for ITU-T/ITU-R/ISO/IEC. A statement of that policy can be found at  
<http://www.itu.int/ITU-T/dbase/patent/patent-policy.html>, with further information available at <http://www.itu.int/ITU-T/ipr/index.html> and in [the ISO/IEC Directives](http://isotc.iso.org/livelink/livelink?func=ll&objId=4230455&objAction=browse&sort=subtype). The form to be used for the formal reporting of patent rights to ITU-T/ITU-R/ISO/IEC can be found at <http://www.itu.int/ITU-T/ipr/index.html>. Contributions to the JCT-VC proposing normative technical content shall contain a non-binding informal notice of whether the submitter may have patent rights that would be necessary for implementation of the resulting standard. The provided informal notice shall indicate the category of anticipated licensing terms according to the ITU-T/ITU-R/ISO/IEC patent statement and licensing declaration form. This obligation to provide an informal notice is supplemental to, and does not replace, any existing obligations of parties with technology included in a final or draft standard to submit formal IPR declarations to ITU-T/ITU-R/ISO/IEC. An example of an informal IPR notification statement for a contribution is provided below.)

**MediaTek Inc. may have current or pending patent rights relating to the technology described in this contribution and, conditioned on reciprocity, is prepared to grant licenses under reasonable and non-discriminatory terms as necessary for implementation of the resulting ITU-T Recommendation | ISO/IEC International Standard (per box 2 of the ITU-T/ITU-R/ISO/IEC patent statement and licensing declaration form).**

# Annex A Proposed Specification for the WD

## Revisions in SPS

|  |  |
| --- | --- |
| seq\_parameter\_set\_rbsp( ) { | Descriptor |
| …… |  |
| interpolation\_filter\_flag | u(1) |
| **seq\_adaptive\_dequantization\_offset\_flag** | u(1) |
| rbsp\_trailing\_bits( ) |  |
| } |  |

**seq\_adaptive\_dequantization\_offset\_flag** equal to 1 specifies that the adaptive de-quantization offset is enabled. Otherwise, the feature is disabled.

## Revisions in Slice Header

|  |  |
| --- | --- |
| slice\_header( ) { | Descriptor |
| …… |  |
| slice\_qp\_delta | se(v) |
| if(seq\_adaptive\_dequantization\_offset\_flag) { |  |
| if(slice\_type==I) { |  |
| **slice\_adaptive\_dequantization\_offset\_luma** | **se(v)** |
| **slice\_adaptive\_dequantization\_offset\_chroma0** | **se(v)** |
| **slice\_adaptive\_dequantization\_offset\_chroma1** | **se(v)** |
| **}** |  |
| **else {** |  |
| **slice\_adaptive\_dequantization\_offset\_luma** | **ue(v)** |
| **slice\_adaptive\_dequantization\_offset\_chroma0** | **ue(v)** |
| **slice\_adaptive\_dequantization\_offset\_chroma1** | **ue(v)** |
| **}** |  |
| **}** |  |
| alf\_param() |  |
| …… |  |
| } |  |

**slice\_adaptive\_dequantization\_offset\_luma** specifiesthenormalized de-quantization offset of luma component in this slice**.** When slice\_adaptive\_dequantization\_offset\_luma is not presented, slice\_adaptive\_dequantization\_offset\_luma shall be inferred as zero.

**slice\_adaptive\_dequantization\_offset\_chroma0** specifiesthenormalized de-quantization offset of the first chroma component, i.e., U component, in this slice**.** When slice\_adaptive\_dequantization\_offset\_ chroma0 is not presented, slice\_adaptive\_dequantization\_offset\_chroma0 shall be inferred as zero.

**slice\_adaptive\_dequantization\_offset\_chroma1** specifiesthenormalized de-quantization offset of the second chroma component, i.e., U component, in this slice**.** When slice\_adaptive\_dequantization\_offset\_ chroma1 is not presented, slice\_adaptive\_dequantization\_offset\_chroma1 shall be inferred as zero.

## Newly proposed syntax elements on signaling cost reduction

To further reduce the bit cost of AQO signaling, additional syntax elements are proposed as follows.

|  |  |
| --- | --- |
| seq\_parameter\_set\_rbsp( ) { | Descriptor |
| …… |  |
| interpolation\_filter\_flag | u(1) |
| **seq\_adaptive\_dequantization\_offset\_flag** | u(1) |
| if(**seq\_adaptive\_dequantization\_offset\_flag**) |  |
| **seq\_aqo\_idx\_flag** | **u(1)** |
| rbsp\_trailing\_bits( ) |  |
| } |  |

**seq\_aqo\_idx\_flag** specifies whether the offset prediction method is used when signaling the adaptive de-quantization offset. When **seq\_aqo\_idx\_flag** is not present, it shall be inferred as zero.

|  |  |
| --- | --- |
| slice\_header( ) { | Descriptor |
| …… |  |
| slice\_qp\_delta | se(v) |
| if(seq\_adaptive\_dequantization\_offset\_flag) { |  |
| if(slice\_type==I) { |  |
| **slice\_adaptive\_dequantization\_offset\_luma** | **se(v)** |
| **slice\_adaptive\_dequantization\_offset\_chroma0** | **se(v)** |
| **slice\_adaptive\_dequantization\_offset\_chroma1** | **se(v)** |
| **}** |  |
| **else {** |  |
| **if(seq\_aqo\_idx\_flag){** |  |
| **slice\_qo\_idx** | **u(3)** |
| **if(0<slice\_qo\_idx && slice\_qo\_idx<7 ) {** |  |
| **slice\_qo\_y\_delta** | **ue(v)** |
| **slice\_qo\_u\_delta** | **ue(v)** |
| **slice\_qo\_v\_delta** | **ue(v)** |
| **}** |  |
| **}** |  |
| **else{** |  |
| **slice\_adaptive\_dequantization\_offset\_luma** | **ue(v)** |
| **slice\_adaptive\_dequantization\_offset\_chroma0** | **ue(v)** |
| **slice\_adaptive\_dequantization\_offset\_chroma1** | **ue(v)** |
| **}** |  |
| **}** |  |
| **}** |  |
| alf\_param() |  |
| …… |  |
| } |  |

**slice\_qo\_idx** specifies the index offset prediction method when signalling the de-quantization offset. Together with **slice\_qo\_y\_delta (***δY***)**, **slice\_qo\_u\_delta (***δU***)** and **slice\_qo\_v\_delta (***δV****)***,the de-quantization offsets *OY, OU* and *OV* are derived as follows.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| slice\_qo\_idx | *N* | Offset of Y component | Offset of U component | Offset of V component |
| 0 | -- | *OY=*0 | *OU=*0 | *OV=*0 |
| 1 | 0 | *OY= δY* | *OU= δU* | *OV= δV* |
| 2 | 1 | *OY=**SY + δY* | *OU= SU + δU* | *OV= SV + δV* |
| 3 | 2 | *OY=2\*SY + δY* | *OU= 2\*SU + δU* | *OV= 2\*SV + δV* |
| 4 | 3 | *OY=3\*SY + δY* | *OU= 3\*SU + δU* | *OV= 3\*SV + δV* |
| 5 | 4 | *OY=4\*SY + δY* | *OU= 4\*SU + δU* | *OV= 4\*SV + δV* |
| 6 | -- | -- | -- | -- |
| 7 | -- | *OY=*64 | *OU=*64 | *OV=*64 |

Where *SY*, *SU*, and *SV* are suggested to be 16, 12, and 12 in the context of HEVC.

## Revisions in Scaling Process for Transform Coefficients

In the current HEVC WD [8], the scaled transform coefficient array dij is derived as follows.

dij = ( cij \* LevelScale(nS)x(nS)[ qP%6 ][ i ][ j ] ) << ( qP/6 + trafoPrecisionExt ), with i, j = 0..nS-1

To incorporate AQO, is modified to .

dij = ( cij \* LevelScale(nS)x(nS)[ qP%6 ][ i ][ j ]+((delta\* LevelScale(nS)x(nS)[ qP%6 ][ i ][ j ])>>7) )

<< ( qP/6 + trafoPrecisionExt ), with i, j = 0..nS-1

Where delta indicates the related de-quantization offset.