

G773 – Higher granularity of quantization parameter scaling and adaptive delta QP signaling

Tammy Lee, Jianle Chen, Jeonghoon Park
Samsung Electronics Co., Ltd

Nov. XX, 2011

Introduction

- ❖ The quantization step size (QStep) of HEVC Test Model
 - QStep increases by approximately 12.25% with each increment of QP
 - QStep doubles when QP is increased by 6
 - QStep mapping is too coarse for fine and accurate rate/quality control
 - QStep mapping is not flexible to fit different application scenario

- ❖ Proposed adaptive delta QP signaling on higher granularity of quantization
 - It is proposed to have new **syntax** to signal **delta QP scale**
 - To support the finer granularity as well as the coarser granularity, the **granularity** of quantization is **doubled**

High granularity QStep mapping (1)

❖ A straightforward extension of current Qstep mapping

- Double QP scaling accuracy
- Qstep increases by approximately 5.95% with each increment of QP
- Qstep doubles when QP is increased by 12

$$Qstep \approx 2^{\frac{QP-4}{6}}, \text{ with } QP = 0 \dots 51 \quad \longrightarrow \quad Qstep \approx 2^{\frac{QP-8}{12}}, \text{ with } QP = 0 \dots 103$$

❖ Quantiser process

- Inverse quantiser

$$Y_{ij} = (Z_{ij} * \text{DeqScale}_{ij}(QP \% 12) \ll (QP/12)) \gg \text{dqbits}$$

$$\text{DeqScale}[12] = \{40, 42, 45, 48, 51, 54, 57, 60, 64, 68, 72, 76\}$$

❖ Advantage of straightforward extension

- Easy adaptation of other modules that is related to QP
 - Deblocking filter
 - CABAC context model initialization
 - Lambda value (Encoder side)

High granularity QStep mapping (2)

❖ Experiment on HM4.0

- Common test configuration QP 44 54 64 74
(equivalent to 22, 27, 32, 37 in HM4.0)
- Additional test with QP-1 43 53 63 73
(equivalent to 21.5, 26.5, 31.5, 36.5 in HM4.0)
- Bit rate increasing (QP-1) vs QP
 - Avg. 18.7 → 9.0% (non-intra configuration)
 - Max. 44.8% → 20.7%

Bit rate increase of QP-1 compared to QP

Test configuration	Bit rate increase			
	HM4.0		Proposed method	
	Average	Max	Average	Max
HE_AI	12.6%	29.2%	6.3%	14.0%
LC_AI	12.4%	25.9%	6.2%	12.5%
HE_RA	18.3%	36.6%	8.8%	17.7%
LC_RA	18.0%	38.6%	8.7%	18.0%
HE_LD	19.6%	44.8%	9.4%	20.7%
LC_LD	19.1%	43.9%	9.1%	20.3%

→ When slice-based QP control is enabled,
the higher granularity supports fine and accurate rate/quality control

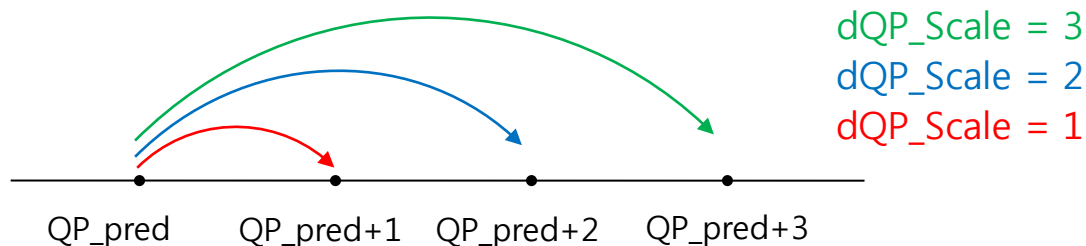
Adaptive delta QP signaling (1)

❖ To support the flexibility between better rate/quality control and dQP bits overhead, new **syntax** to signal **delta QP scale** is proposed

- Slice level QP can be kept as highest level always
- QP of current CU calculated as

$$QP_curr = QP_pred + dQP * dQP_Scale$$

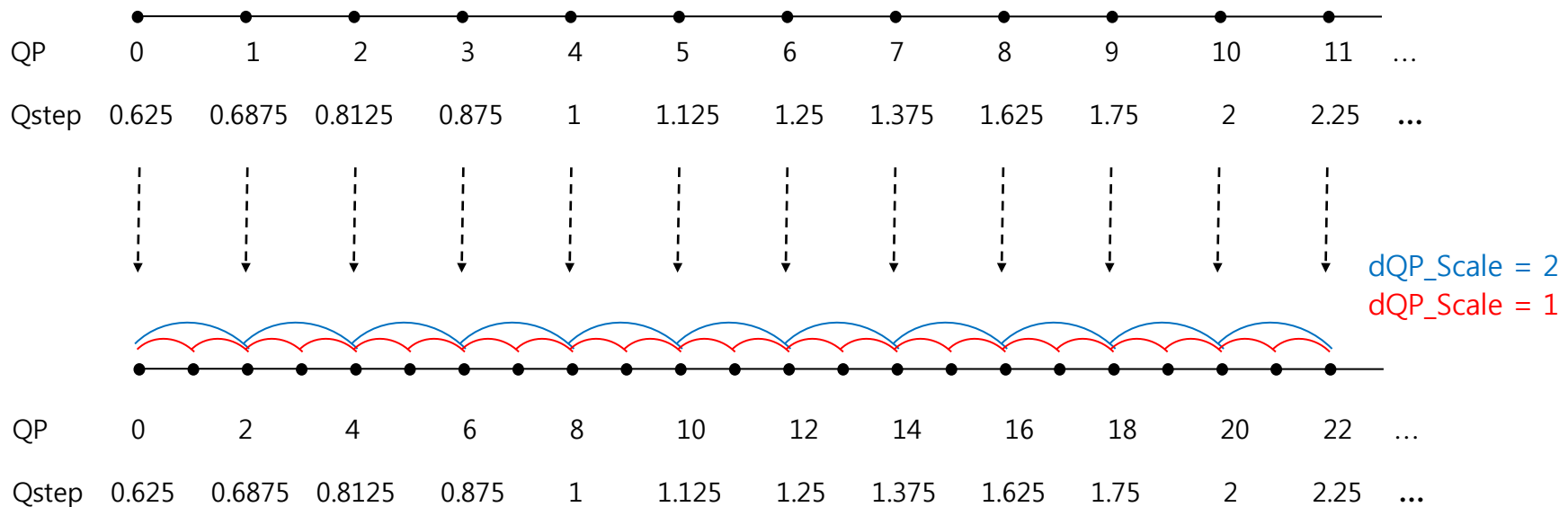
- dQP_Scale controls CU-level QP update speed
- dQP=1 with different dQP_Scale
 - dQP_Scale = 1 → Qstep increase 5.95%
 - dQP_Scale = 2 → Qstep increase 12.25%
 - dQP_Scale = 3 → Qstep increase 18.92%



dQP=1 with different dQP_Scale

Adaptive delta QP signaling (2)

- ❖ To emulate current HEVC QP mapping solution,
 - Set Slice QP = even value (: doubled value of 22, 27, 32, 37) and $dQP_scale = 2$
- ❖ Example of CU-level QP update speed
 - Slice QP = even and $dQP_Scale = 1$ or 2



Adaptive delta QP signaling (3)

❖ Experiment on CE4 subtest 1 software

- The software is used to evaluate bits overhead for coding dQP
- Floating QP (fQP) of the software is rounded as

$$QP_curr = Round \left\{ \frac{(2 \times fQP - QP_pred)}{dQP_Scale} \times dQP_Scale \right\}$$

- dQP is signalled after it is divided by dQP_Scale
 - This results in lower overhead in case of coarse QP_curr by dQP_Scale=3

dQP bits increase compared to CE4 subtest 1

Test configuration	dQP_Scale == 1		dQP_Scale == 3	
	Y BD-rate	dQP Bits increase	Y BD-rate	dQP Bits increase
HE_AI	1.3%	31.1%	-0.6%	-17.0%
LC_AI	1.6%	35.7%	-0.6%	-17.4%
HE_RA	1.2%	29.6%	-0.6%	-16.5%
LC_RA	1.5%	34.8%	-0.7%	-17.6%
HE_LD	1.4%	29.1%	-0.7%	-16.4%
LC_LD	1.9%	34.5%	-0.9%	-17.6%

Summary

- ❖ Double the QStep granularity of HEVC
 - 12.25% -> 5.95% QStep increase with each QP increment
 - Avg. 18.7% -> 9.0% bit increase with each QP increment

- ❖ Adaptive delta QP signalling with dQP_Scale in slice header
 - Control CU level QStep/QP update speed
 - Trade-off of better rate/quality control and dQP bits overhead
 - Slice level high QP mapping granularity
 - Easy adaptation of other modules that is related to QP

Thank you !