



AHG4/AHG9: Syntax for restricting slices and tiles

Chih-Wei Hsu, Chia-Yang Tsai , Yu-Wen Huang, Yung-Chang Chang, Chih-Ming Wang,
Chia-Yun Cheng, Shawmin Lei



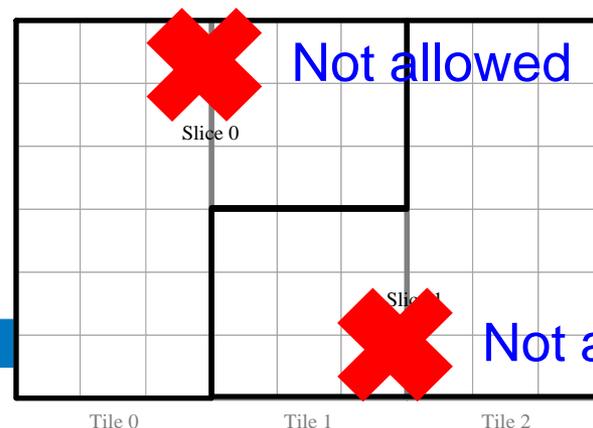
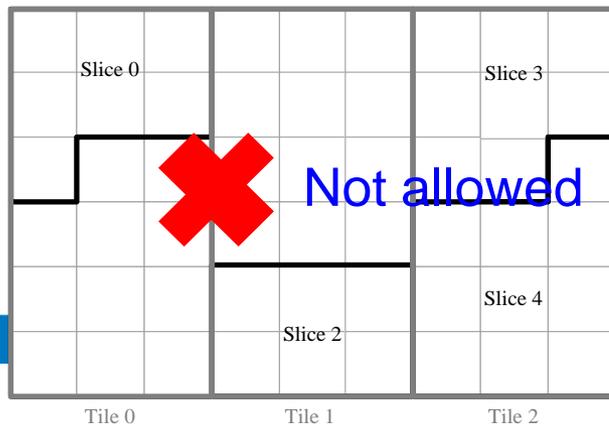
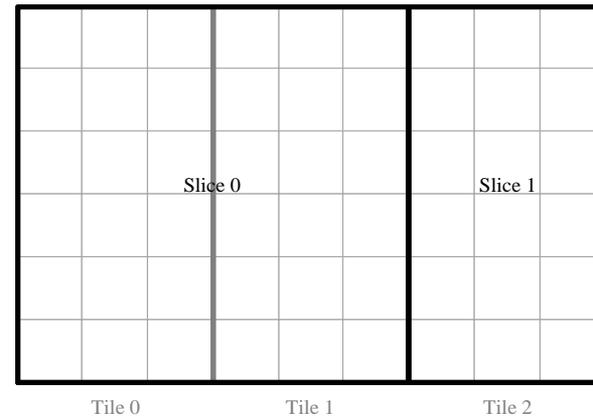
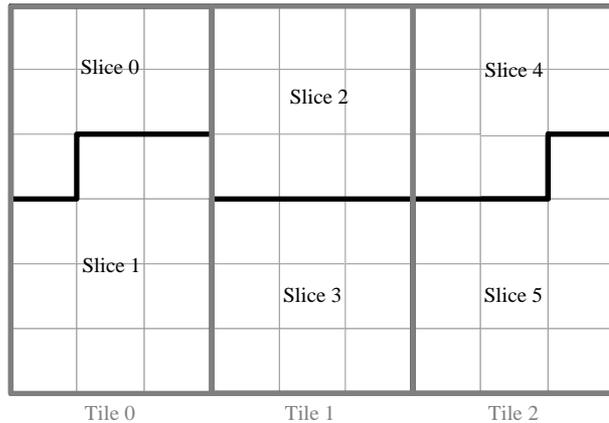
Presented by Shawmin Lei
10th JCT-VC Meeting in Stockholm, SE
11–20 July 2012

Overall Summary

- In HEVC, both slice and tile coding structures can coexist.
 - With normative constraint on the encoder side
 - All slices within a tile shall be complete
 - All tiles within a slice shall be complete
- Practical decoders may still have to deal with bitstreams that unintentionally violate the constraint.
- In this contribution, conditions to signal `end_of_slice_flag` are changed to better guide an encoder to follow the constraint
 - Unintentional violation of the constraints can be avoided
- Simulation results reportedly show no bit rate increases and unchanged run time for the proposed syntax changes

Introduction

- In HEVC, slice and tile coding structures can coexist
- With constraints to limit the coexistence of slices and tiles
 - All slices within a tile shall be complete
 - All tiles within a slice shall be complete

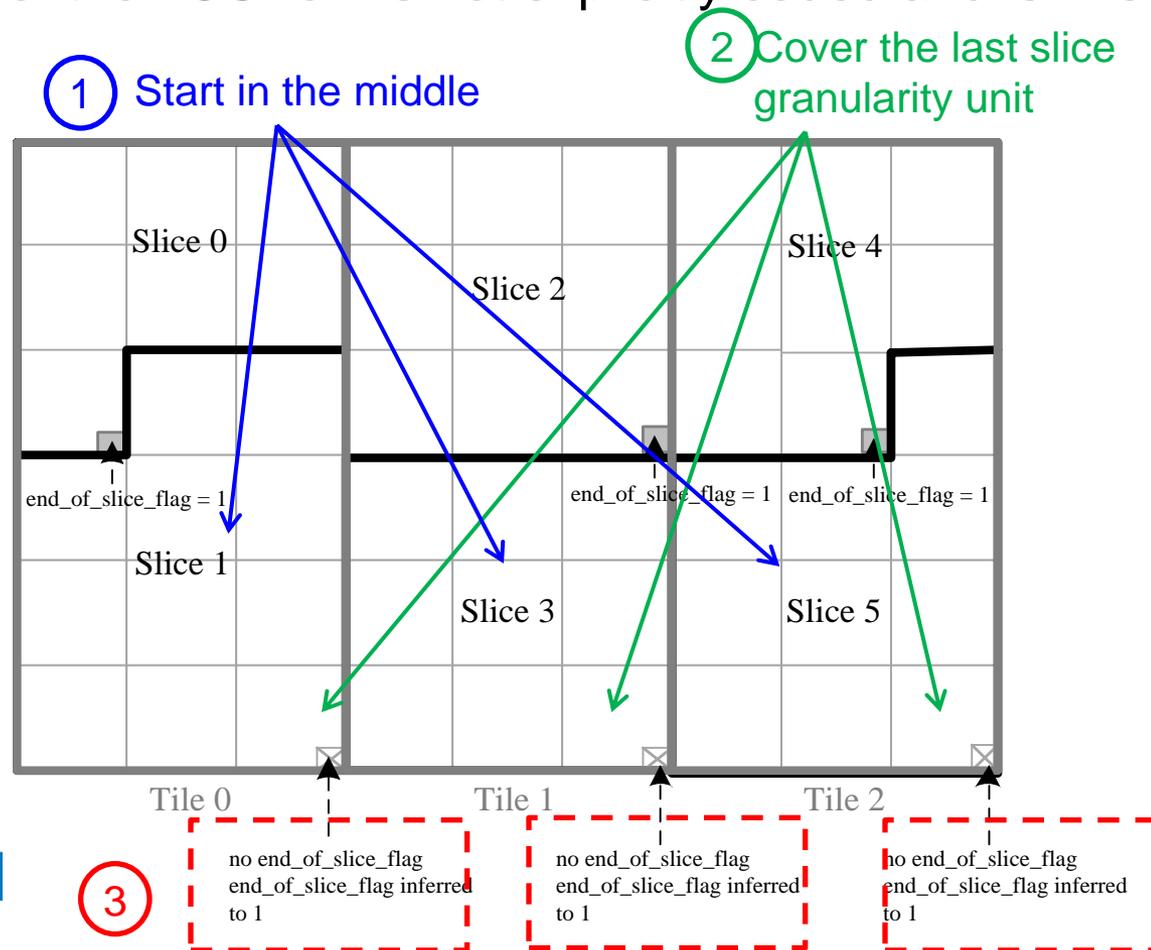


Problem Definition

- Only constraint on encoder side, no normative change on bistream syntax
- Decoder is likely to receive illegal bitstreams
 - Decoder complexity is increased in order to handle illegal bitstreams in practical product
- Propose to embed the constraint to syntax
 - Avoid inadvertent violation of the restriction by encoders
 - Relieve decoders from having to deal with illegal bitstreams

Proposed Method

- If slice starts in the middle of a tile
 - The `end_of_slice_flag` of the last slice granularity unit (LCU or CU) of the LCU row is not explicitly coded and is inferred as true

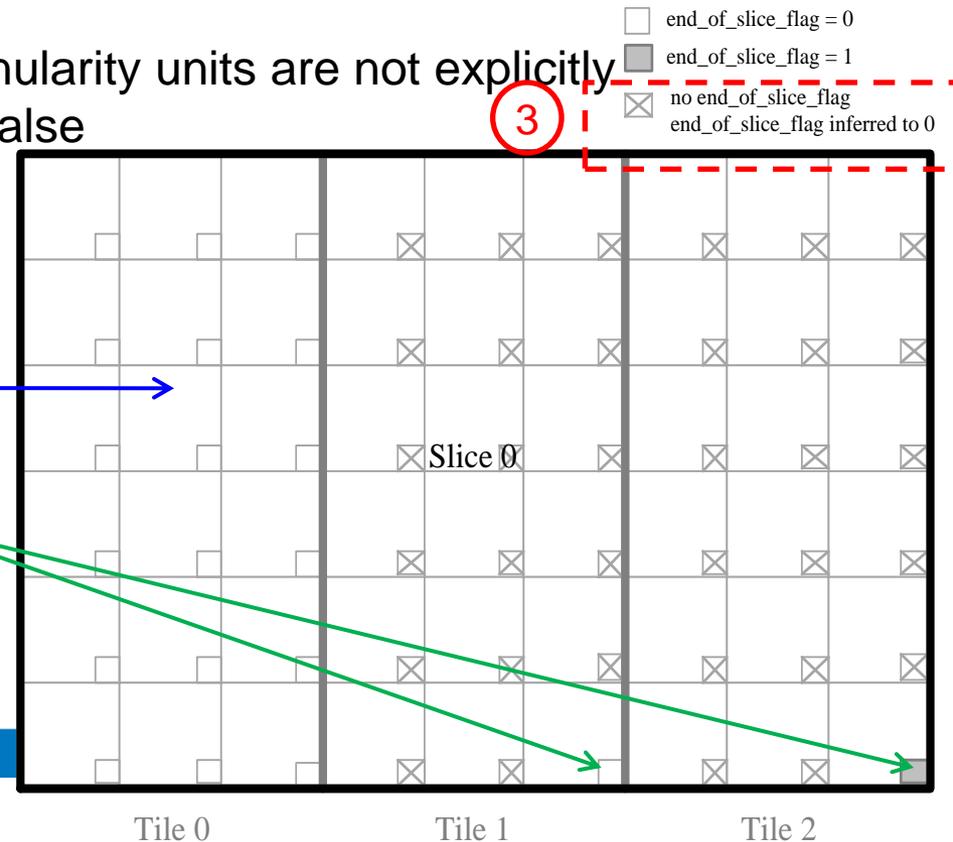


Proposed Method (cont'd)

- If tiles contained in slice
 - In the first tile, `end_of_slice_flag` are coded as usual
 - In each of the rest tiles in a picture,
 - Only the `end_of_slice_flag` of the last slice granularity unit is explicitly coded
 - Those of non-last slice granularity units are not explicitly coded and are inferred as false

① First tile coded as usual

② Only `end_of_slice_flag` for the last slice granularity unit



Simulation Results: Configuration 1

- Anchor: HM-7.0-dev, revision 2458
- Test tiles containing complete slices
 - **UniformSpacingIdc: 1; NumTileColumnsMinus1 : 1; NumTileRowsMinus1: 0**
 - Two tile columns
 - **SliceMode: 1; SliceArgument: 4**
 - Maximum 4 LCU per slice
- AI-Main, RA-Main, LB-Main are tested
 - No significant impact on BD-rates and run time

	All Intra Main		
	Y	U	V
Class A	0.0%	0.0%	0.0%
Class B	0.0%	0.0%	0.0%
Class C	0.0%	0.0%	0.0%
Class D	0.0%	0.0%	0.0%
Class E	0.0%	0.0%	0.0%
Overall	0.0%	0.0%	0.0%
	0.0%	0.0%	0.0%
Class F	0.0%	0.0%	0.0%
Enc Time[%]	100%		
Dec Time[%]	103%		

	Random Access Main		
	Y	U	V
Class A	0.0%	0.0%	0.0%
Class B	0.0%	0.0%	0.0%
Class C	0.0%	0.0%	0.0%
Class D	0.0%	0.0%	0.0%
Class E			
Overall	0.0%	0.0%	0.0%
	0.0%	0.0%	0.0%
Class F	0.0%	0.0%	0.0%
Enc Time[%]	100%		
Dec Time[%]	90%		

	Low delay B Main		
	Y	U	V
Class A			
Class B	0.0%	0.0%	0.0%
Class C	0.0%	0.0%	0.0%
Class D	0.0%	0.0%	0.0%
Class E	0.0%	0.0%	0.0%
Overall	0.0%	0.0%	0.0%
	0.0%	0.0%	0.0%
Class F	0.0%	0.0%	0.0%
Enc Time[%]	100%		
Dec Time[%]	100%		

Simulation Results: Configuration 2

- Anchor: HM-7.0-dev, revision 2458
- Test slices containing complete tiles
 - **UniformSpacingIdc**: 1; **NumTileColumnsMinus1** : 2; **NumTileRowsMinus1**: 2
 - nine tiles (3x3)
 - **SliceMode**: 3; **SliceArgument**: 4
 - Maximum 4 tiles per slice
- AI-Main, RA-Main, LB-Main are tested
 - No significant impact on BD-rates and run time

	All Intra Main		
	Y	U	V
Class A	0.0%	0.0%	0.0%
Class B	0.0%	0.0%	0.0%
Class C	0.0%	0.0%	0.0%
Class D	0.0%	0.0%	0.0%
Class E	0.0%	0.0%	0.0%
Overall	0.0%	0.0%	0.0%
	0.0%	0.0%	0.0%
Class F	0.0%	0.0%	0.0%
Enc Time[%]	100%		
Dec Time[%]	99%		

	Random Access Main		
	Y	U	V
Class A	0.0%	0.1%	0.0%
Class B	0.0%	0.0%	0.0%
Class C	0.0%	0.0%	0.1%
Class D	0.0%	-0.1%	-0.1%
Class E			
Overall	0.0%	0.0%	0.0%
	0.0%	0.0%	0.0%
Class F	0.0%	0.1%	0.1%
Enc Time[%]	100%		
Dec Time[%]	99%		

	Low delay B Main		
	Y	U	V
Class A			
Class B	0.0%	0.0%	0.2%
Class C	0.0%	-0.1%	0.0%
Class D	0.0%	0.3%	0.3%
Class E	0.0%	0.0%	-0.7%
Overall	0.0%	0.0%	0.0%
	0.0%	0.1%	0.1%
Class F	-0.1%	0.4%	0.4%
Enc Time[%]	100%		
Dec Time[%]	100%		

Conclusions

- In this contribution, normative syntax changes for the constraints on the coexistence of tiles and slices are proposed
- With the proposed syntax changes, encoders are guided to generate bitstreams conforming to the constraints
- Decoders can be relieved from the possible burden that has to deal with the illegal bitstreams that violate the restrictions inadvertently
- Simulation results reportedly show the proposed method has no impact on coding efficiency or run time

MEDIA/TEK

Thank you

