

JVET-P0534

AHG16/Non-CE5: On deblocking at ALF virtual boundaries

Zhipin Deng, Li Zhang, Kai Zhang, Hongbin Liu, Hsiao Chiang Chuang



Summary

■ Problem statement

- In VTM-5.0, the in-loop filtering can be applied in a CTU pipelined way of 64x60, 64x64, 64x64...
- In VTM-6.0, the deblocking can be applied at 4x4 luma grid
 - Samples below the ALF VB shall be deblocked together with samples above the ALF VB
 - However, the ALF filter of those samples still need to be postponed until lower CTU comes
 - The previous CTU pipelined in-loop filtering is influenced

■ Proposal

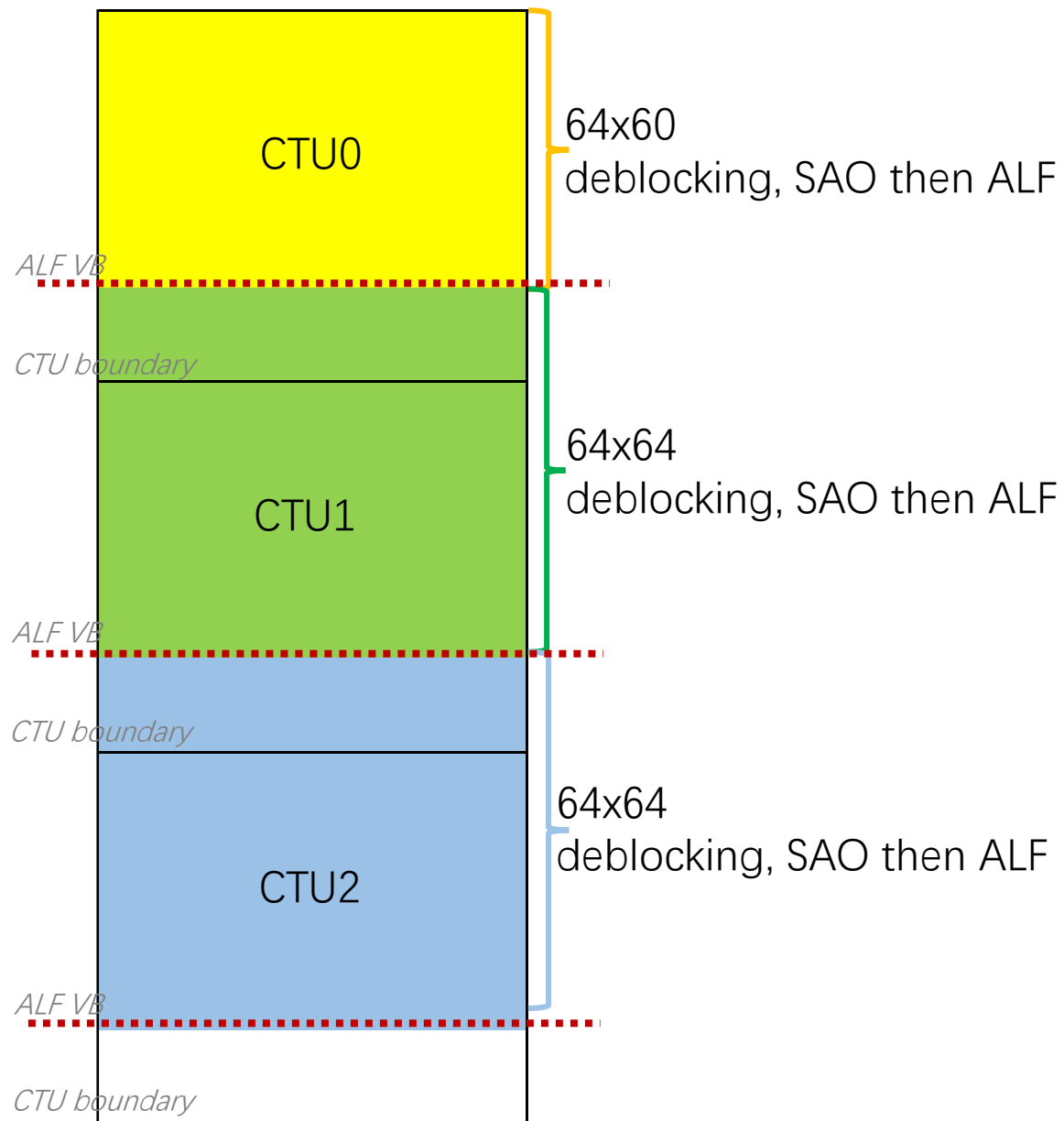
- Disable the 4x4 grid luma deblocking for edges coinciding with ALF VB

■ Experimental results

BD-rate Y	AI	RA	LDB	LDP
CTC	0.00%	0.01%	0.00%	0.03%
ALF off	0.00%	0.01%	0.02%	-0.01%

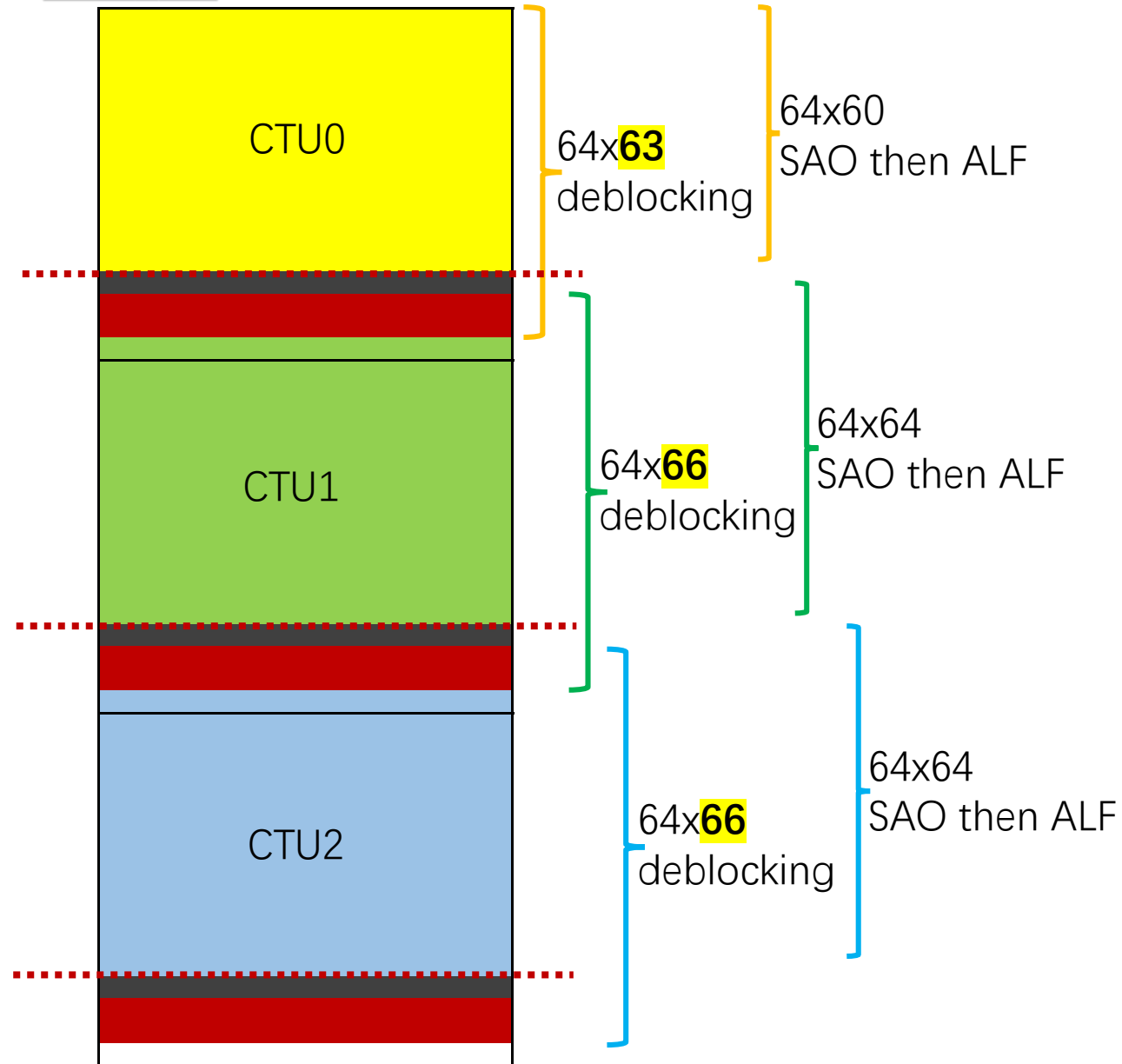
CTU pipelined processing

VTM-5.0, and the proposed



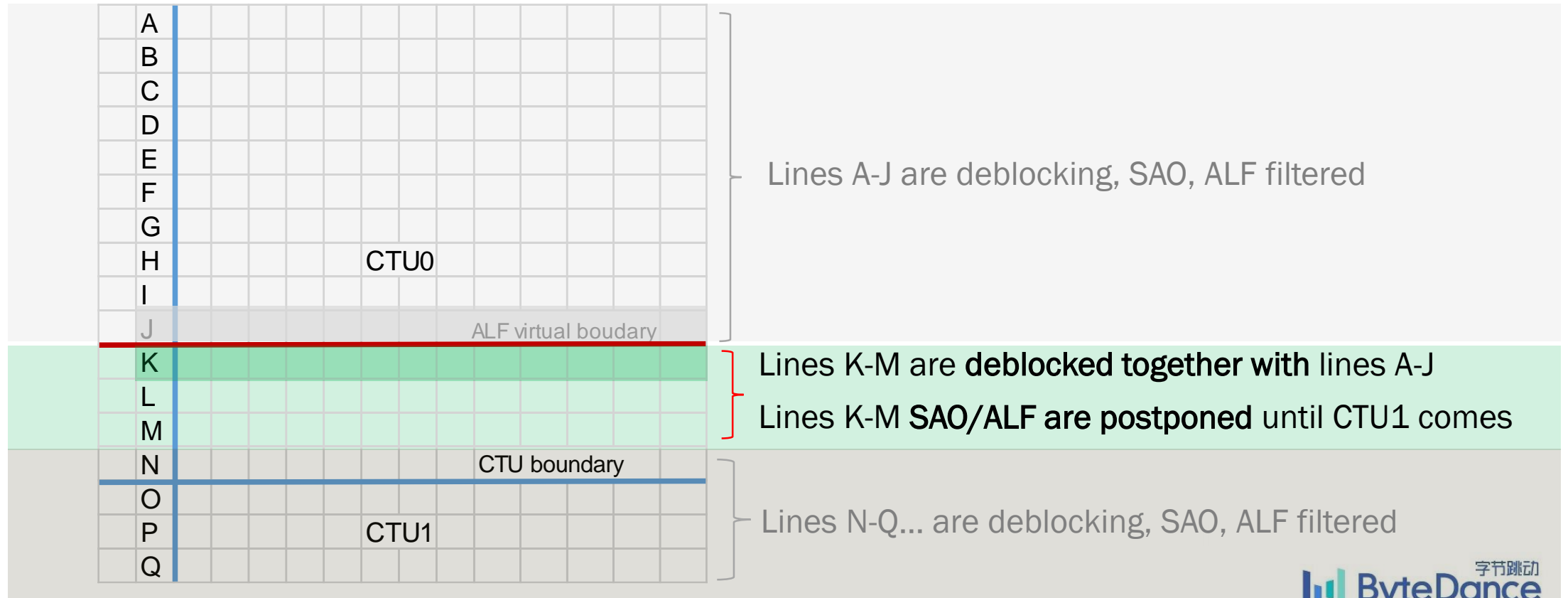
VTM-6.0

64



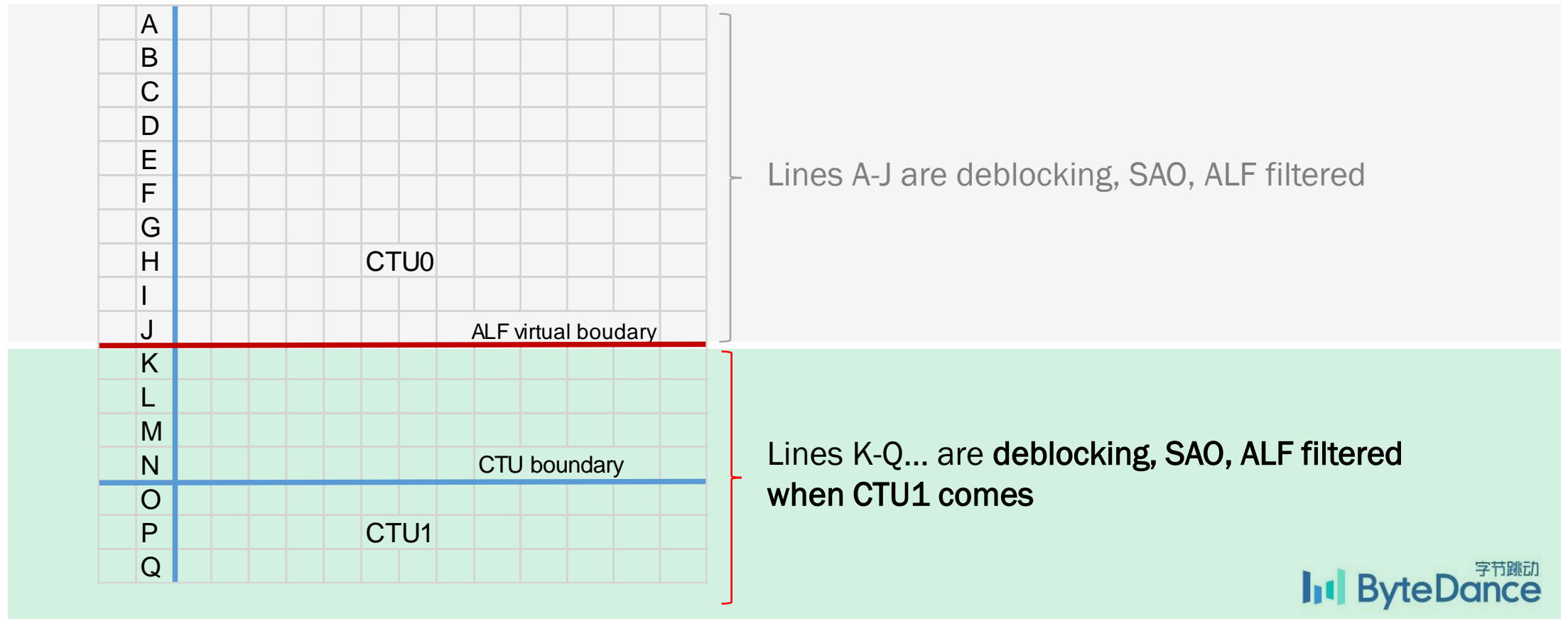
CTU pipelined in-loop filtering in VTM-6.0

- When a deblocking edge coincides with the ALF VB
 - Lines K-M shall be **deblocked together** with samples above ALF VB
 - Lines K-M **SAO/ALF are postponed** until CTU1 comes
 - Lead to inconsistent CTU pipelined processing



Proposal

- Proposed to disable the 4x4 grid luma deblocking for edges coinciding with ALF VB
 - To make a consistent CTU pipelined processing



Conclusions

■ Benefits

- Keep a consistent CTU pipelined processing for deblocking, SAO and ALF
- Minor compression impact

BD-rate Y	AI	RA	LDB	LDP
CTC	0.00%	0.01%	0.00%	0.03%
ALF off	0.00%	0.01%	0.02%	-0.01%

- Might not be critical in visual quality
 - affect just at most two lines per CTU

■ It is recommended to adopt the proposed method

Thanks Kwai for crosschecking! (JVET-P0727)