

Temporal merging candidate derivation with reduced complexity

JCTVC-I0116

Minhua Zhou

(Texas Instruments Inc., USA)

Summary of Proposal

1. Proposal

- Fixing reference picture index to zero for TMVP in merging and AMVP candidate list derivation process

2. Motivations

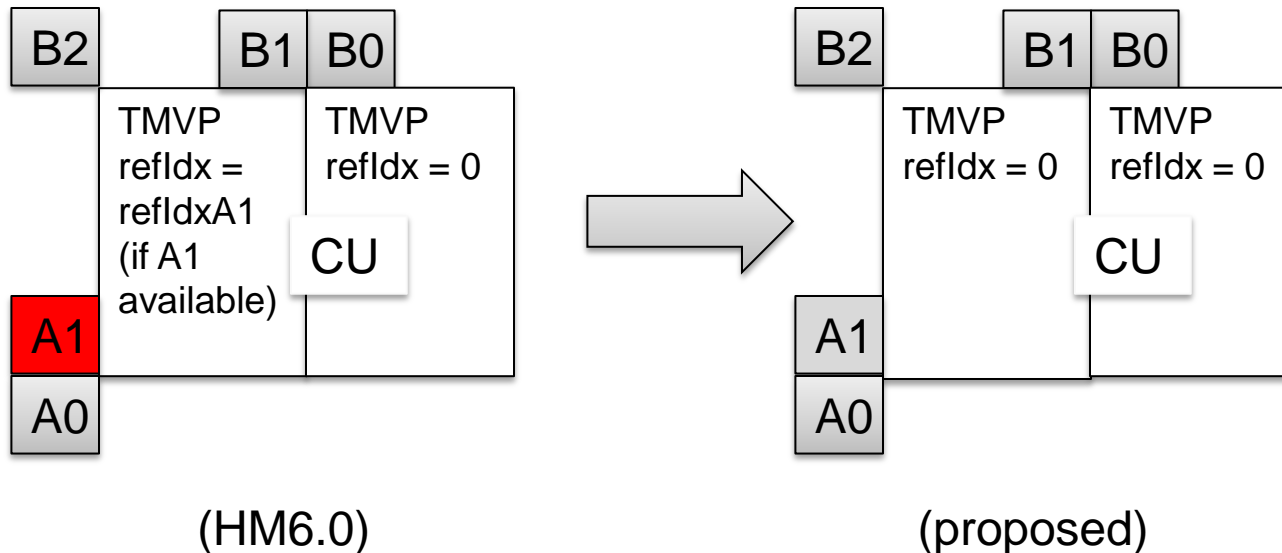
- Improve cache efficiency for merge motion estimation
- Improve throughput of merging and AMVP candidate list derivation by decoupling TMVP derivation from the rest of process
- Facilitate pre-calculation of TMVPs for a LCU by leveraging motion data compression

3. Coding efficiency impact

- None

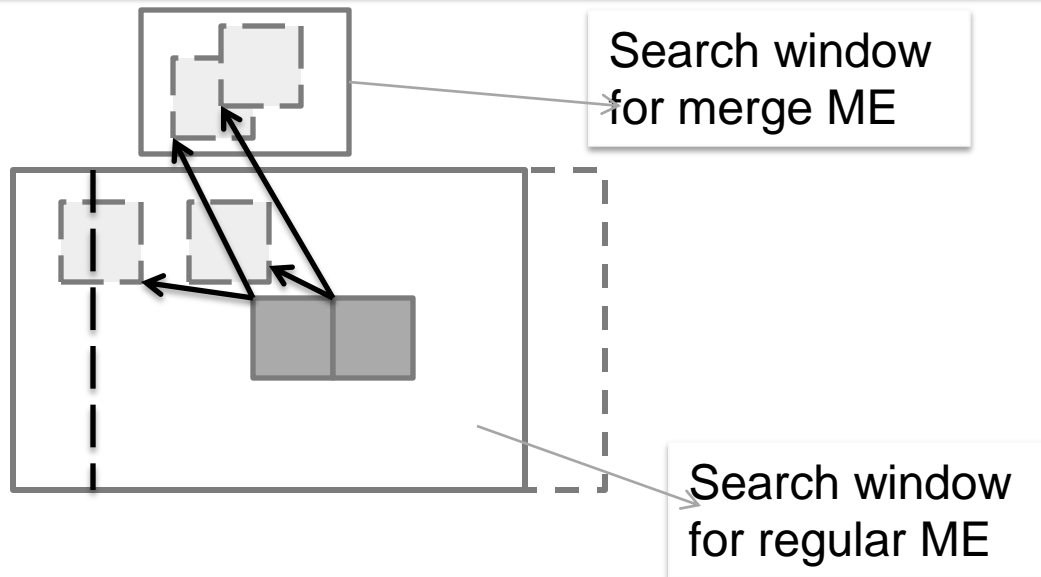
HM6.0 TMVP refIdx Derivation

1. In HM6.0 TMVP refIdx depends on spatial neighbor A1
2. Such a design has several drawbacks
 - Increased bandwidth requirements for merge motion estimation
 - Reduced throughput for merging/AMVP list derivation process
 - Make pre-calculation of TMVPs difficult

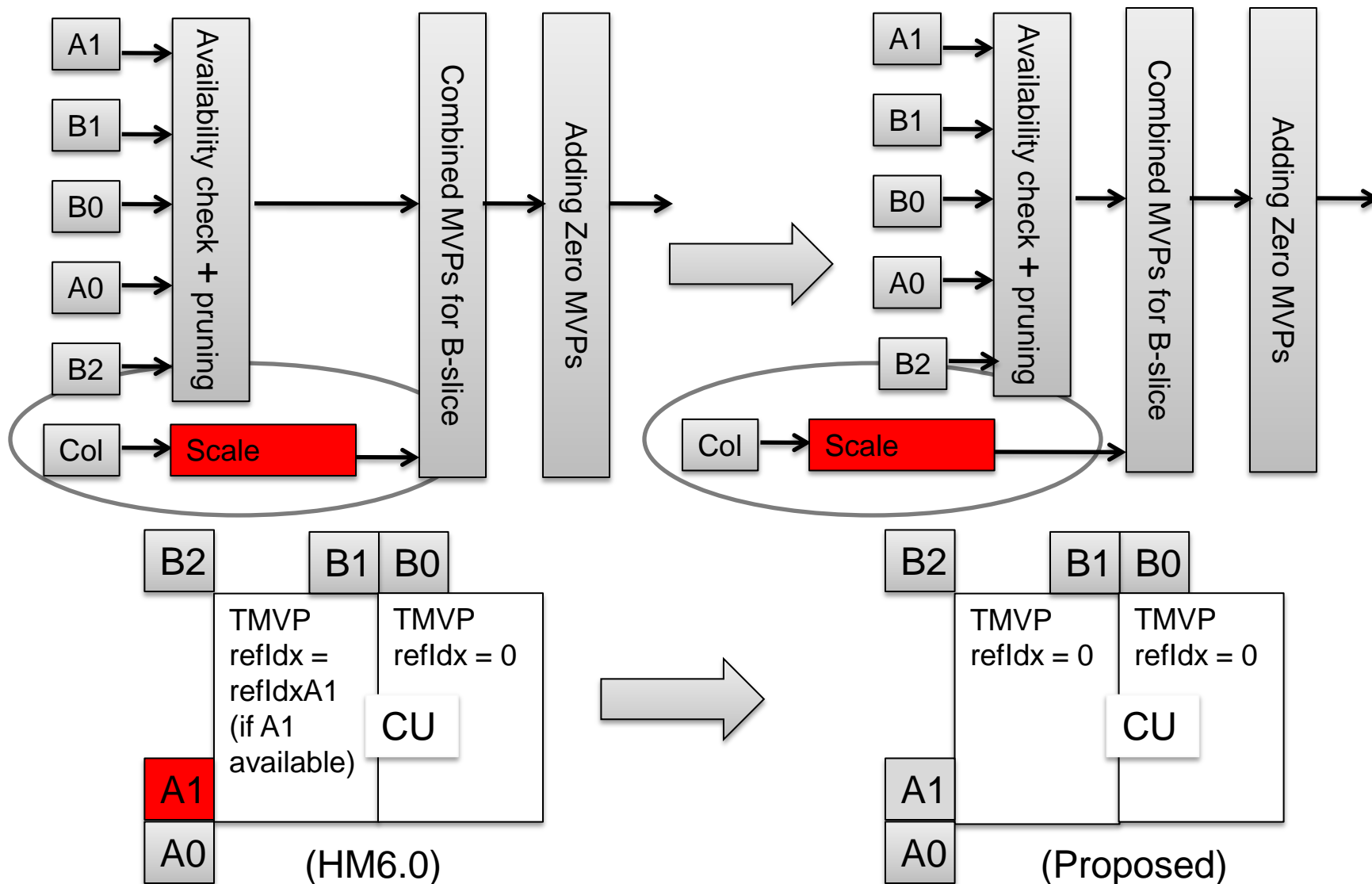


Cache efficiency for merge ME

1. For merge motion estimation TMVP oftentimes goes beyond the major search window
 - Separate memory cache is needed for buffering reference block for merge/skip search
 - Changing refIdx for TMVP from PU to PU inevitably reduces cache efficiency and increases memory bandwidth



Throughput Difference (e.g. merging list)



Pre-calculation of TMVPs

1. In HM6.0 co-located motion data are down-sampled
2. A smart encoder or decoder can leverage this property and pre-calculate TMVPs for a LCU instead of computing them on individual PU basis
3. Not fixing TMVP refIdx to zero magnifies the number of combinations for the TMVP scaling , and makes the pre-calculation of TMVPs difficult

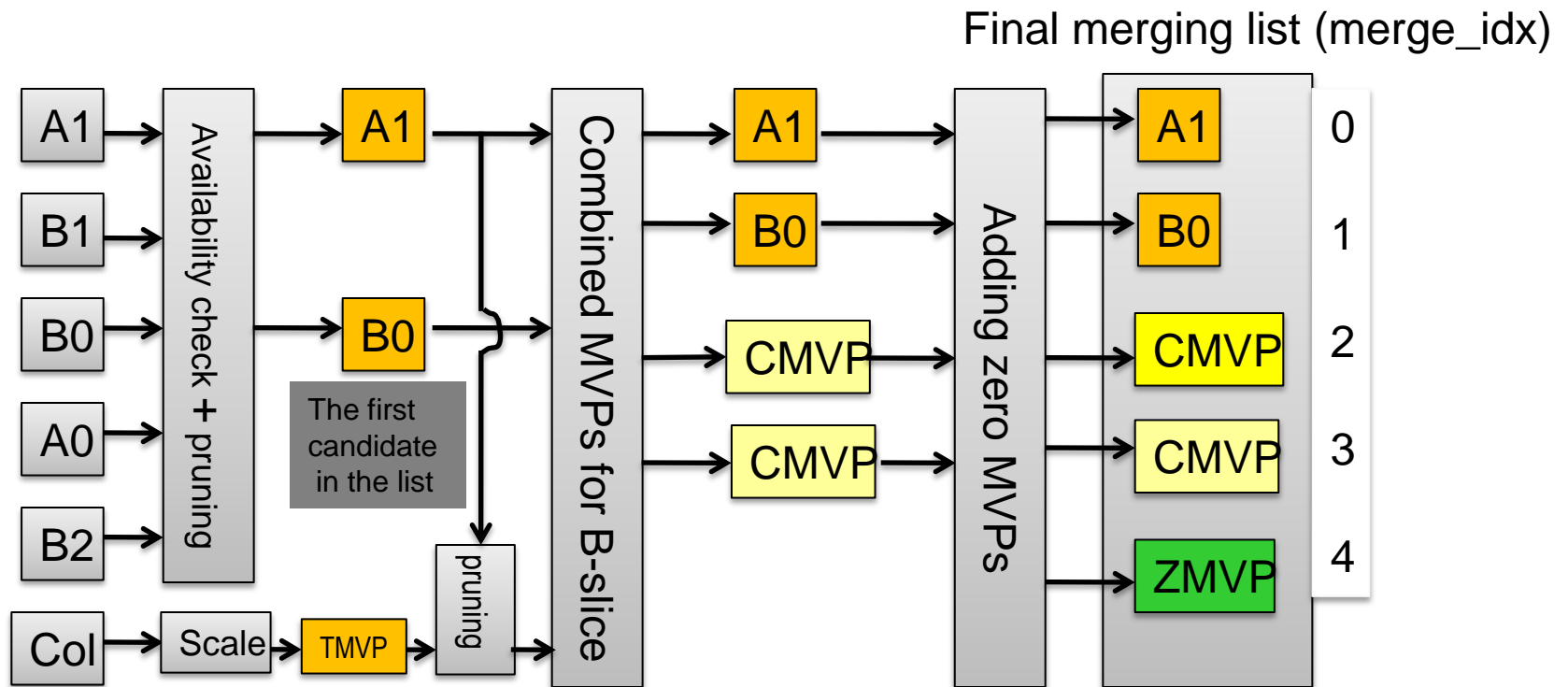
Experimental Results

1. There is up to 0.1% loss when TMVP refidx is fixed to 0
 - In HM6.0 TMVP is appended to the merging candidate list w/o pruning
 - Fixing the TMVP reference index to 0 increases the likelihood that the TMVP is duplicated relative to one of the spatial merging candidates in the list

Setting	RA- Main	RA- HE10	LB- Main	LB- HE10	LP- Main	LP- HE10
Overall(%)	0.0	0.0	0.1	0.0	0.0	0.0

TMVP Pruning Process

- The TMVP is appended to the list only if the TMVP has different motion data from the first merging candidate in the list.



Experimental Results

Setting	RA-Main	RA-HE10	LB-Main	LB-HE10	LP-Main	LP-HE10
TMVP reldx = 0	0.0	0.0	0.1	0.0	0.0	0.0
TMVP pruning	0.0	0.0	-0.1	-0.1	-0.1	-0.1
TMP reldx = 0 plus pruning	0.0	0.0	0.0	0.0	0.0	-0.1

TMVP pruning leads to 0.4% gain in class E of LP-Main

Conclusions

1. Fixing TMVP refIdx to 0 is beneficial in several ways
 - lowering memory bandwidth requirements of merge motion estimation
 - Improving throughput of merging/AMVP list derivation by decoupling the TMVP derivation from the rest of process at PU-level
 - Enabling TMVP derivation with reduced complexity by leveraging the motion data down-sampling to pre-calculate TMVPs on 16x16 grids instead of individual PUs
2. Adding a simple TMVP pruning process on top of fixing TMVP refIdx to 0 leads to slight gains
3. It is recommended to adopt both changes into HM

Thanks to Microsoft for cross-check (JCTVC-I0411)