

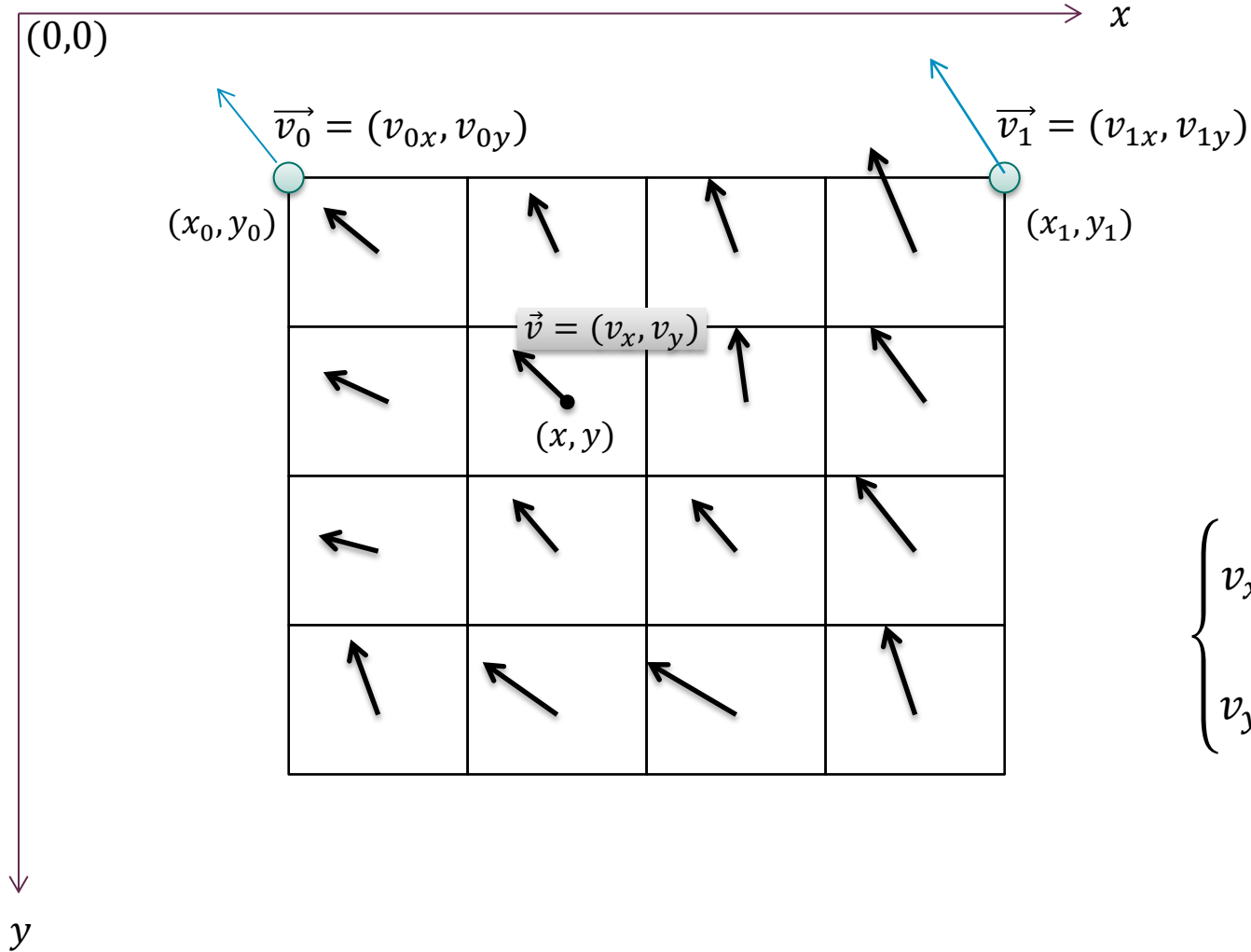
A STUDY ON THE AFFINE MERGE MODE (JVET-K0052)



Minhua Zhou and Brian Heng

Broadcom Inc.

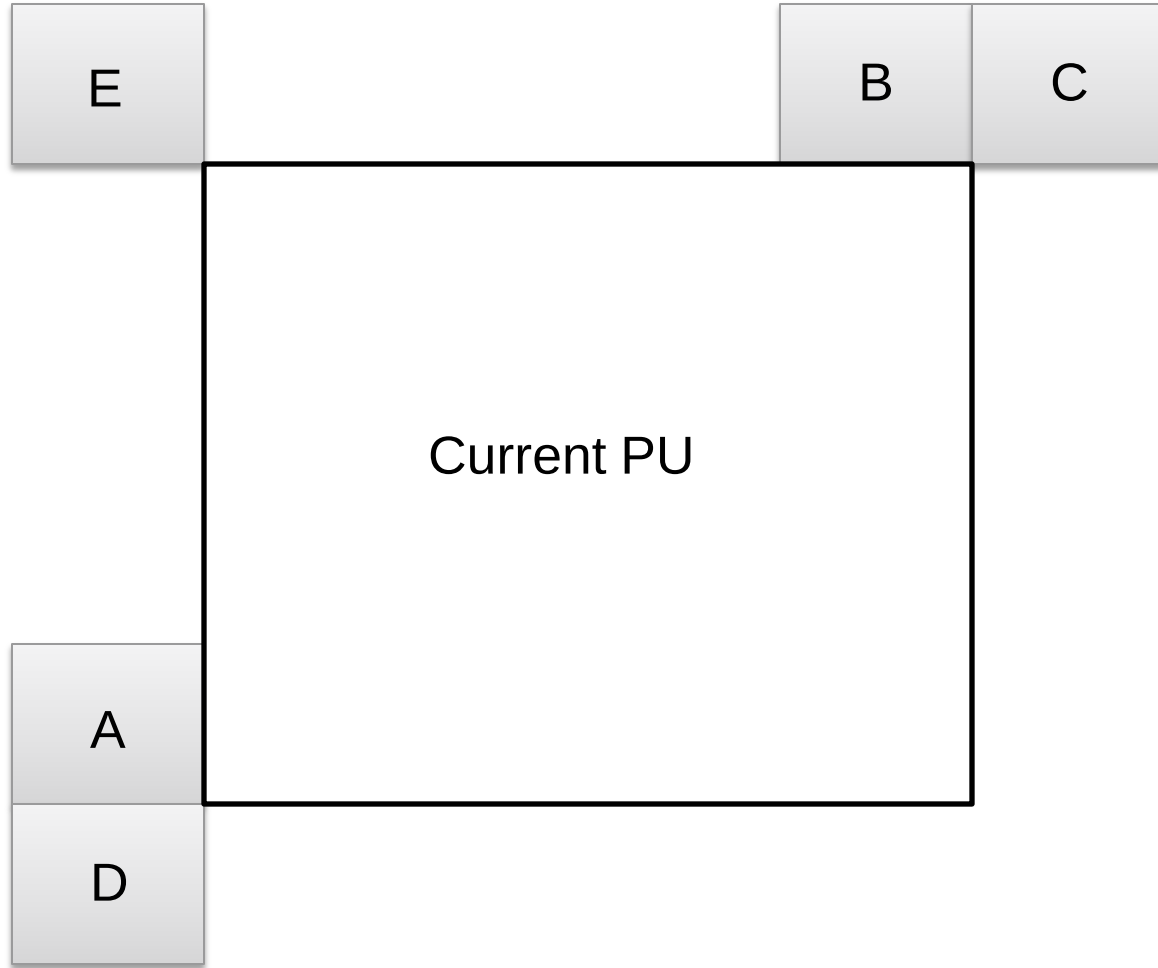
BMS1.0 Affine Mode



■ Sub-block vector derivation

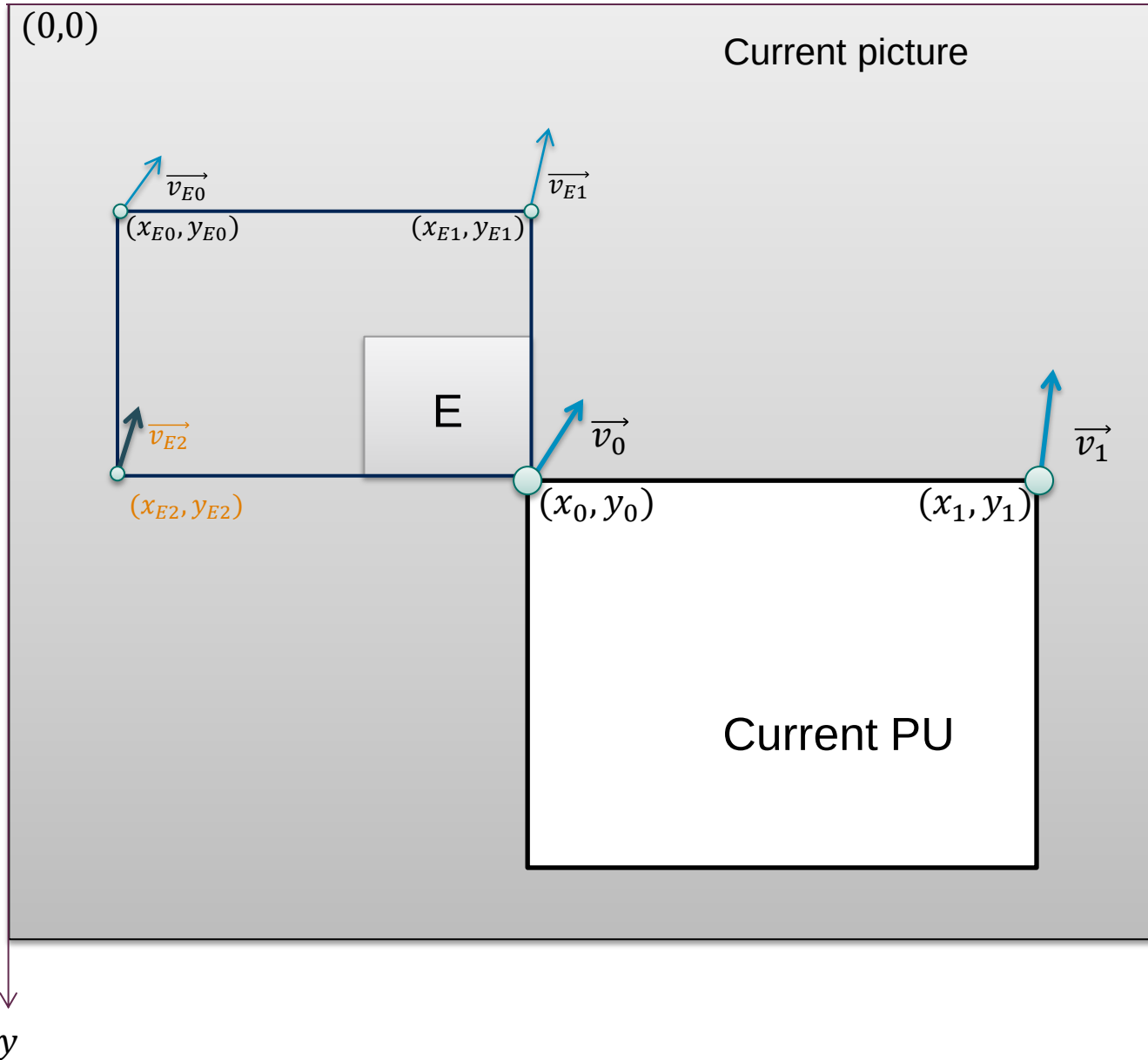
$$\begin{cases} v_x = \frac{(v_{1x} - v_{0x})}{(x_1 - x_0)} (x - x_0) - \frac{(v_{1y} - v_{0y})}{(x_1 - x_0)} (y - y_0) + v_{0x} \\ v_y = \frac{(v_{1y} - v_{0y})}{(x_1 - x_0)} (x - x_0) + \frac{(v_{1x} - v_{0x})}{(x_1 - x_0)} (y - y_0) + v_{0y} \end{cases}$$

BMS1.0 Affine Merge Mode



- A PU can inherit the affine motion data from the first available neighboring PU using the affine mode, in the order of A, B, C, D and E

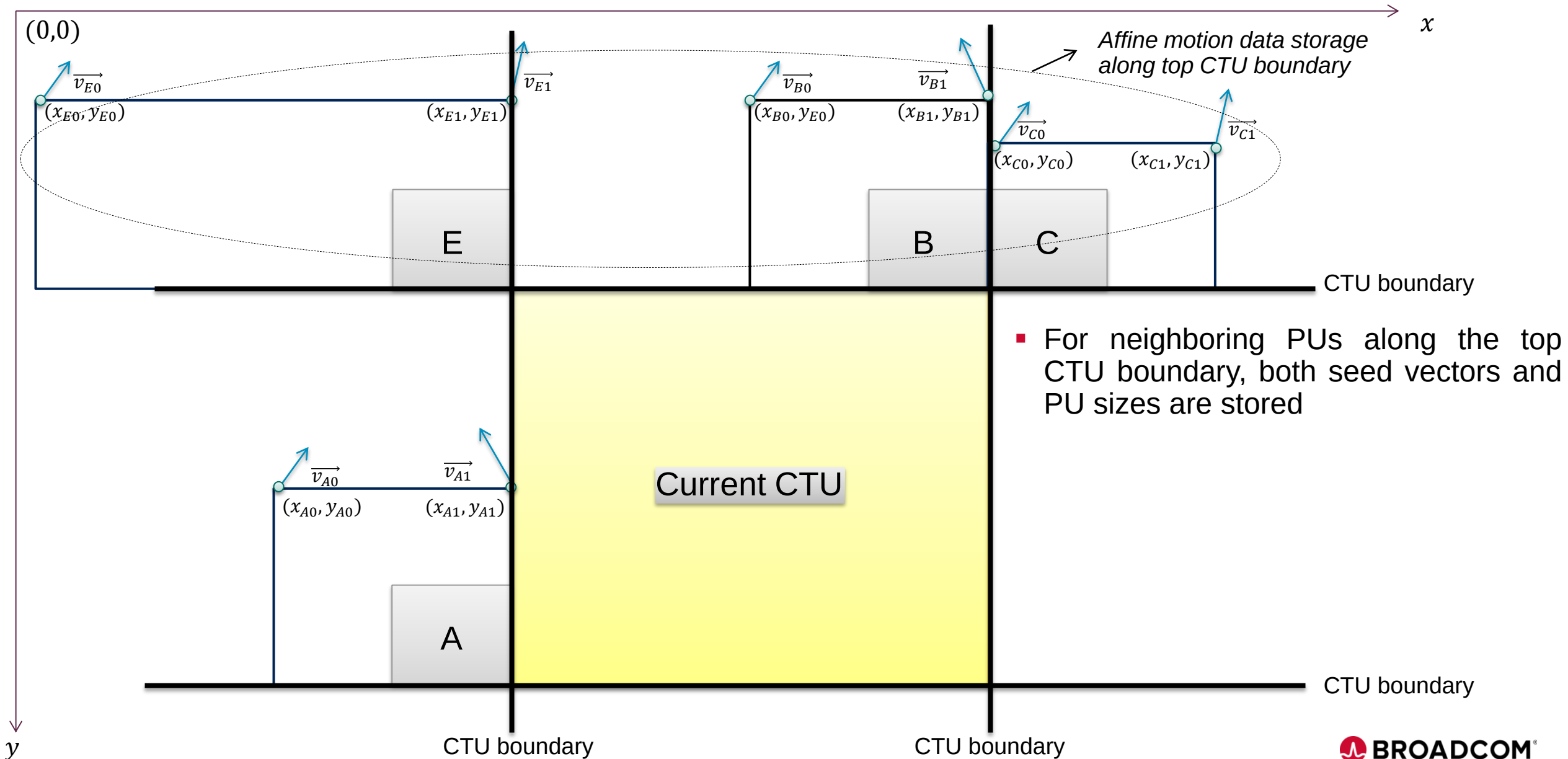
Seed Vector Derivation in Affine Merge Mode of BMS1.0



- In the BM1.0, the seed vectors of the current PU is NOT derived by using the 4-parameter model.
- Instead, the bottom-right seed vector of the neighboring PU, $\vec{v}_{E2}$, is computed using the 4-parameter model.
- The seed vectors of the current PU are computed using the 6-parameter model, and by

$$\begin{cases} v_{0x} = \frac{(v_{E1x} - v_{E0x})}{(x_{E1} - x_{E0})} (x_0 - x_{E0}) + \frac{(v_{E2x} - v_{E0x})}{(y_{E2} - y_{E0})} (y_0 - y_{E0}) + v_{E0x} \\ v_{0y} = \frac{(v_{E1y} - v_{E0y})}{(x_{E1} - x_{E0})} (x_0 - x_{E0}) + \frac{(v_{E2y} - v_{E0y})}{(y_{E2} - y_{E0})} (y_0 - y_{E0}) + v_{E0y} \\ v_{1x} = \frac{(v_{E1x} - v_{E0x})}{(x_{E1} - x_{E0})} (x_1 - x_{E0}) + \frac{(v_{E2x} - v_{E0x})}{(y_{E2} - y_{E0})} (y_1 - y_{E0}) + v_{E0x} \\ v_{1y} = \frac{(v_{E1y} - v_{E0y})}{(x_{E1} - x_{E0})} (x_1 - x_{E0}) + \frac{(v_{E2y} - v_{E0y})}{(y_{E2} - y_{E0})} (y_1 - y_{E0}) + v_{E0y} \end{cases}$$

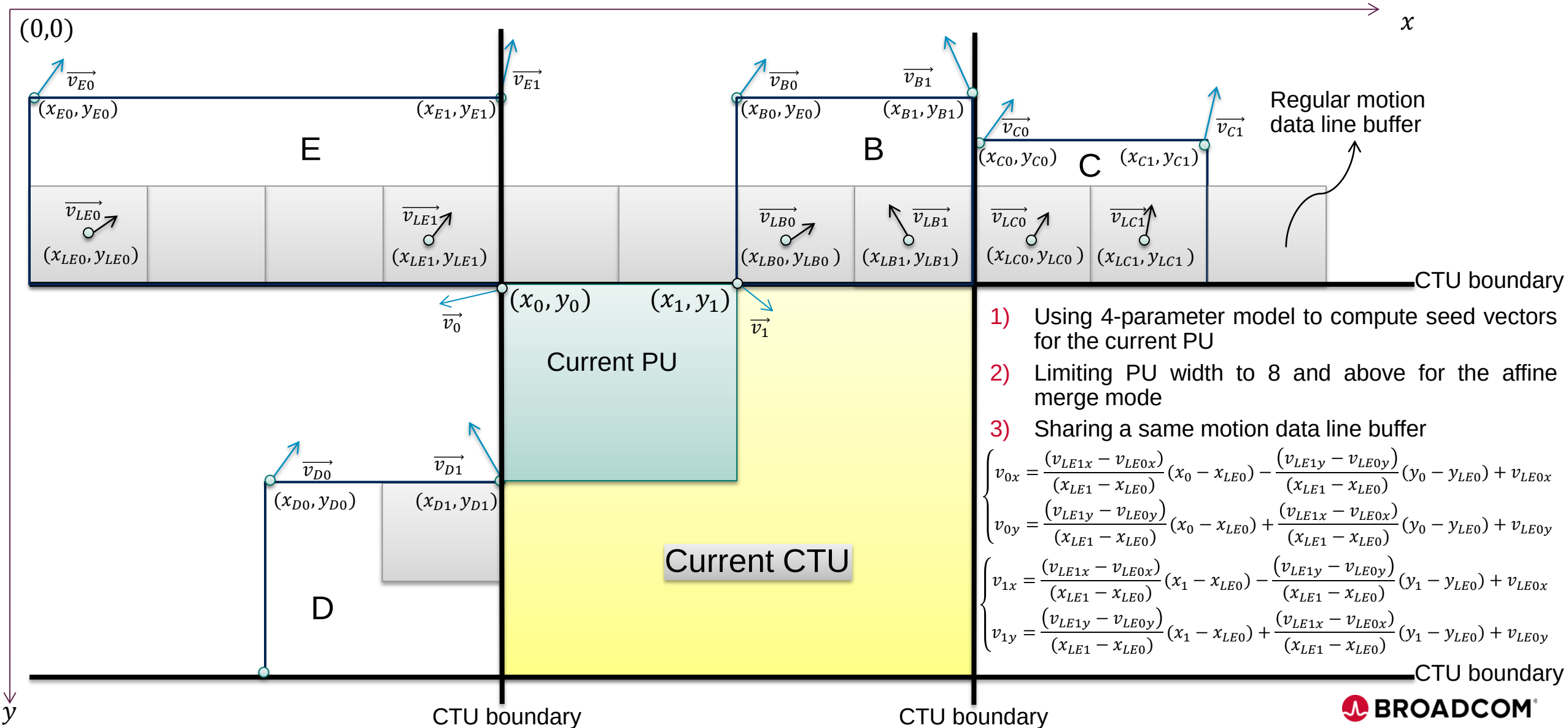
Line Buffer Needed for the Affine Merge Mode



Motion Data Line Buffer Size

Luma picture width = 4096		line buffer for merge/skip and AMVP mode		line buffer for affine merge mode in BMS1.0	
	Bits/unit	Number of units per 4 luma samples	list0 & list1 line buffer size (bytes)	Number of units per 4 luma samples	list0 & list1 line buffer size (bytes)
Motion/seed vectors	32	2	8192	4	16384
Reference index	4	2	1024	2	1024
PU width	5	0	0	1	640
PU height	5	0	0	1	640
Total (bytes)			9216		18688

Proposed Simplifications



Reduction of Motion Data Line Buffer Size

Luma picture width = 4096		line buffer for merge/skip and AMVP mode		line buffer for affine merge mode in BMS1.0		line buffer for affine merge mode after simplification	
	Bits/unit	Number of units per 4 luma samples	list0 & list1 line buffer size (bytes)	Number of units per 4 luma samples	list0 & list1 line buffer size (bytes)	Number of units per 4 luma samples	list0 & list1 line buffer size (bytes)
Motion/seed vectors	32	2	8192	4	16384	0	0
Reference index	4	2	1024	2	1024	0	0
PU width	5	0	0	1	640	0.5	320
PU height	5	0	0	1	640	0	0
Total (bytes)			9216		18688		320

Tests Conducted using BMS1.0 Software

- **The following five tests were performed for BMS configurations:**
 - 1) **Affine merge disabled:** disabling the affine merge mode in BMS1.0.
 - 2) **4-parameter model:** using 4-parameter based affine motion model to directly compute the seed vectors for the current PU of affine merge mode.
 - 3) **Min affine merge PU width 8:** test 2) plus disabling affine merge mode for PUs whose width is less than 8.
 - 4) **High-precision LB sharing:** test 3) plus line buffer sharing (MVs stored in high precision in the motion data line buffer).
 - 5) **Low-precision LB sharing:** test 3) plus line buffer sharing (MVs stored in low precision in the motion data line buffer).
- **The following test was also performed for BMS in VTM configurations but with “Affine” and “HighPrecMv” on:**
 - 6) **Low-precision LB sharing (VTM):** test 3) plus line buffer sharing (MVs stored in low precision in the motion data line buffer).

Experimental Results (Test 1 & Test 2)

	Test 1) Affine merge disabled					Test 2) 4-parameter model				
	Random Access Main 10					Random Access Main 10				
	Y	U	V	EncT	DecT	Y	U	V	EncT	DecT
Class A1	0.37%	0.11%	0.11%	100%	97%	0.01%	-0.13%	-0.03%	97%	95%
Class A2	0.90%	0.69%	0.73%	87%	86%	0.02%	-0.01%	0.00%	94%	93%
Class B	0.57%	0.42%	0.50%	87%	85%	0.00%	-0.02%	-0.02%	93%	92%
Class C	0.33%	0.29%	0.14%	91%	87%	-0.01%	0.02%	0.04%	95%	90%
Overall	0.53%	0.38%	0.37%	91%	88%	0.00%	-0.03%	0.00%	94%	92%
Class D	0.58%	0.34%	0.24%	94%	88%	0.03%	0.06%	-0.07%	99%	93%
Class F (optional)	0.32%	0.30%	0.31%	99%	96%	-0.01%	-0.04%	-0.06%	102%	97%
	Low Delay B Main 10					Low Delay B Main 10				
	Y	U	V	EncT	DecT	Y	U	V	EncT	DecT
Class B	0.43%	0.15%	0.66%	95%	93%	-0.04%	-0.31%	0.06%	100%	99%
Class C	0.34%	0.27%	0.27%	93%	87%	-0.05%	-0.05%	-0.07%	85%	81%
Class E	0.10%	0.53%	0.14%	100%	103%	-0.21%	0.38%	0.60%	105%	105%
Overall	0.32%	0.28%	0.40%	95%	93%	-0.09%	-0.05%	0.15%	96%	94%
Class D	1.05%	1.14%	1.04%	98%	97%	0.02%	0.32%	0.20%	93%	92%
Class F (optional)	0.18%	0.47%	0.02%	97%	94%	0.07%	0.50%	0.06%	116%	114%
	Low Delay P Main 10					Low Delay P Main 10				
	Y	U	V	EncT	DecT	Y	U	V	EncT	DecT
Class B	0.43%	0.40%	0.34%	100%	102%	-0.06%	-0.01%	-0.10%	103%	104%
Class C	0.41%	0.49%	0.40%	102%	104%	-0.03%	0.17%	-0.21%	93%	94%
Class E	0.41%	1.06%	0.19%	90%	92%	-0.16%	-0.09%	-0.36%	91%	92%
Overall	0.42%	0.59%	0.32%	98%	100%	-0.07%	0.03%	-0.20%	96%	98%
Class D	1.02%	1.51%	0.27%	99%	96%	-0.02%	0.69%	-0.45%	94%	85%
Class F (optional)	0.30%	0.70%	0.27%	100%	93%	-0.17%	0.19%	-0.03%	98%	94%

Experimental Results (Test 3 & Test 4)

	Test 3) Min affine merge PU width 8					Test 4) High-precision LB sharing				
	Random Access Main 10					Random Access Main 10				
	Y	U	V	EncT	DecT	Y	U	V	EncT	DecT
Class A1	0.02%	-0.07%	0.02%	92%	90%	0.03%	-0.08%	-0.01%	102%	98%
Class A2	0.00%	0.03%	0.02%	87%	86%	-0.01%	0.02%	0.02%	98%	96%
Class B	-0.02%	0.02%	-0.07%	91%	92%	-0.01%	-0.01%	-0.01%	92%	91%
Class C	0.01%	0.04%	-0.12%	96%	92%	0.05%	-0.01%	-0.05%	95%	89%
Overall	0.00%	0.01%	-0.04%	92%	90%	0.01%	-0.02%	-0.01%	96%	93%
Class D	0.01%	0.09%	-0.01%	96%	87%	0.04%	0.08%	0.12%	94%	85%
Class F (optional)	-0.02%	-0.06%	-0.06%	102%	101%	-0.02%	-0.02%	-0.07%	100%	100%
	Low Delay B Main 10					Low Delay B Main 10				
	Y	U	V	EncT	DecT	Y	U	V	EncT	DecT
Class B	-0.02%	-0.27%	-0.11%	93%	93%	0.02%	-0.26%	0.41%	104%	108%
Class C	-0.03%	-0.05%	-0.19%	82%	80%	-0.01%	0.06%	-0.12%	93%	92%
Class E	-0.23%	0.19%	-0.70%	116%	116%	-0.21%	-0.31%	0.00%	113%	120%
Overall	-0.07%	-0.08%	-0.29%	94%	93%	-0.05%	-0.16%	0.13%	102%	105%
Class D	-0.03%	-0.04%	-0.04%	109%	106%	0.04%	0.32%	-0.35%	104%	114%
Class F (optional)	-0.02%	0.34%	0.11%	118%	120%	-0.03%	1.12%	0.48%	103%	103%
	Low Delay P Main 10					Low Delay P Main 10				
	Y	U	V	EncT	DecT	Y	U	V	EncT	DecT
Class B	-0.03%	0.11%	-0.04%	106%	108%	-0.01%	0.06%	-0.10%	102%	100%
Class C	0.03%	0.10%	-0.13%	97%	97%	0.01%	0.01%	0.00%	97%	96%
Class E	-0.13%	-0.34%	-0.97%	87%	86%	-0.05%	0.51%	-0.19%	90%	88%
Overall	-0.04%	-0.01%	-0.30%	98%	99%	-0.02%	0.15%	-0.09%	97%	96%
Class D	-0.09%	0.07%	0.08%	88%	82%	0.00%	0.02%	-0.06%	90%	81%
Class F (optional)	-0.04%	-0.13%	0.21%	107%	98%	-0.02%	0.28%	-0.18%	93%	86%

Experimental Results (Test 5 & Test 6)

	Test 5) Low-Precision LB sharing					Test 6) Low-Precision LB sharing (VTM)				
	Random Access Main 10					Random Access Main 10				
	Y	U	V	EncT	DecT	Y	U	V	EncT	DecT
Class A1	0.04%	-0.16%	-0.01%	100%	95%	0.00%	-0.09%	-0.06%	105%	101%
Class A2	0.03%	0.00%	-0.02%	95%	94%	0.02%	-0.01%	0.01%	106%	105%
Class B	0.02%	0.03%	0.02%	92%	90%	0.03%	-0.01%	0.07%	98%	98%
Class C	0.03%	0.05%	-0.02%	91%	83%	-0.03%	-0.01%	-0.02%	99%	104%
Overall	0.03%	-0.01%	-0.01%	94%	90%	0.00%	-0.03%	0.01%	101%	101%
Class D	0.04%	0.11%	-0.06%	99%	88%	-0.02%	-0.03%	-0.01%	101%	98%
Class F (optional)	-0.01%	-0.06%	-0.03%	98%	86%	0.00%	0.04%	-0.02%	93%	91%
	Low Delay B Main 10					Low Delay B Main 10				
	Y	U	V	EncT	DecT	Y	U	V	EncT	DecT
Class B	-0.01%	-0.33%	0.04%	98%	94%	0.03%	-0.12%	0.09%	99%	101%
Class C	0.02%	0.10%	-0.06%	81%	76%	0.02%	0.01%	0.16%	90%	88%
Class E	-0.27%	0.05%	0.11%	104%	106%	-0.01%	0.15%	0.42%	107%	103%
Overall	-0.07%	-0.09%	0.03%	93%	90%	0.02%	-0.01%	0.20%	98%	97%
Class D	0.03%	0.10%	-0.02%	96%	94%	0.09%	0.38%	0.29%	102%	93%
Class F (optional)	-0.06%	0.24%	0.15%	101%	103%	-0.10%	0.48%	0.76%	99%	100%
	Low Delay P Main 10					Low Delay P Main 10				
	Y	U	V	EncT	DecT	Y	U	V	EncT	DecT
Class B	-0.01%	0.11%	0.14%	97%	98%	0.04%	0.05%	-0.13%	99%	101%
Class C	0.02%	0.03%	-0.09%	97%	96%	-0.02%	-0.06%	-0.08%	107%	117%
Class E	-0.10%	-0.25%	-0.47%	86%	88%	0.18%	-0.09%	0.19%	93%	93%
Overall	-0.03%	-0.01%	-0.09%	94%	95%	0.06%	-0.02%	-0.04%	100%	104%
Class D	0.01%	-0.63%	-0.17%	92%	88%	-0.07%	0.12%	0.30%	89%	91%
Class F (optional)	0.03%	-0.15%	-0.44%	91%	86%	0.12%	0.20%	0.37%	95%	101%

Recommendation

- It is recommended to study the simplification of the affine (merge) mode to make it more implementation friendly.
- Potential areas for improvements (comments based on BMS1.0 reference code)
 - 1) Line buffer reduction
 - 2) Removal of floating point operations and divisions
 - 3) Removal of redundant sub-block motion vector derivations
 - see `setAllAffineMv( )` for deriving vector context
 - see `xPredAffineBlk( )` “almost” same math for deriving actual MC vectors

This two-stage process also currently has bugs when the current PU is 4xN or Nx4 (half of the 4 corner seed vectors get overwritten since there are only 2 corners in these small blocks).
 - 4) High precision vs. low precision storage of affine sub-block motion vectors
 - 5) Memory bandwidth impact
 - 4x4 bi-pred is likely a problem (may not be specific to the affine (merge) mode)