ON LINE BUFFER REDUCTION FOR AFFINE MODE (JVET-L0046)

Minhua Zhou

Broadcom Inc.

BROADCOM

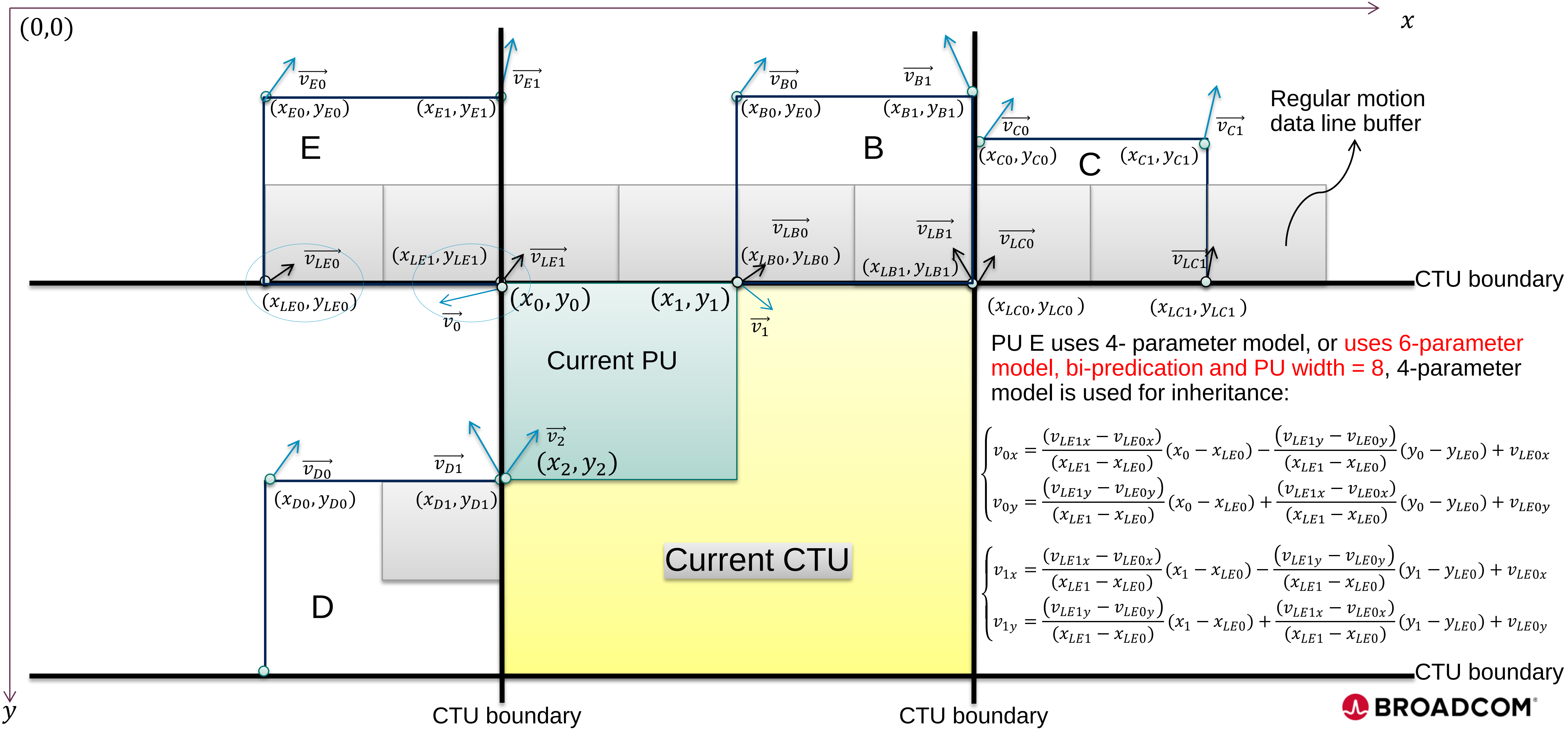
Background

- Line buffer reduction algorithms tested in CE4.1.11 (see JVET-L0045) led to some outliers (e.g. peak loss 0.34% in sequence CatRobot1 of RA in CE4.1.11 test 1)
- The tested algorithms in CE4.1.11 use the 4-parameter for the affine inheritance along the top CTU boundary regardless whether the source neighboring PU uses 4- or 6-parmeter model
  - Cutting corners when the source neighboring PU uses 6-parameter model
- Changes are suggested in this contribution to improve coding efficiency without increasing line buffer size when compared to the algorithms tested in CE4.1.11

Overview of Proposed Algorithm

- For the affine motion data inheritance from a neighboring PU located at the top CTU boundary,
  - If the neighboring PU uses the 4-parameter model, or the neighboring PU has PU width 8, uses the 6-parameter model and bi-directional prediction, the 4-parameter model using the bottom-left and bottom-right control point motion vectors (CPMVs) is applied to the inheritance.
  - Otherwise, if the neighboring PU uses the 6-parameter model and unidirectional prediction, the 6-parameter model using the bottom-left, bottom-right and top-left CPMVs is applied to the inheritance.
  - Otherwise, if the neighboring PU uses the 6-parameter model and unidirectional prediction but its PU width is larger than 8, the 6-parameter model using the bottom-left, bottom-right CPMVs and bottom-middle two sub-block vectors is applied to the inheritance.
- Corner cutting only happens when the neighboring PU uses 6-parameter model and bi-directional prediction, and has PU width 8.
  - In all the other cases, mathematically equivalent inheritance (to the current VTM2.0.1 method) is used

Case 1 (PU E uses bi-prediction, and PU width = 8)



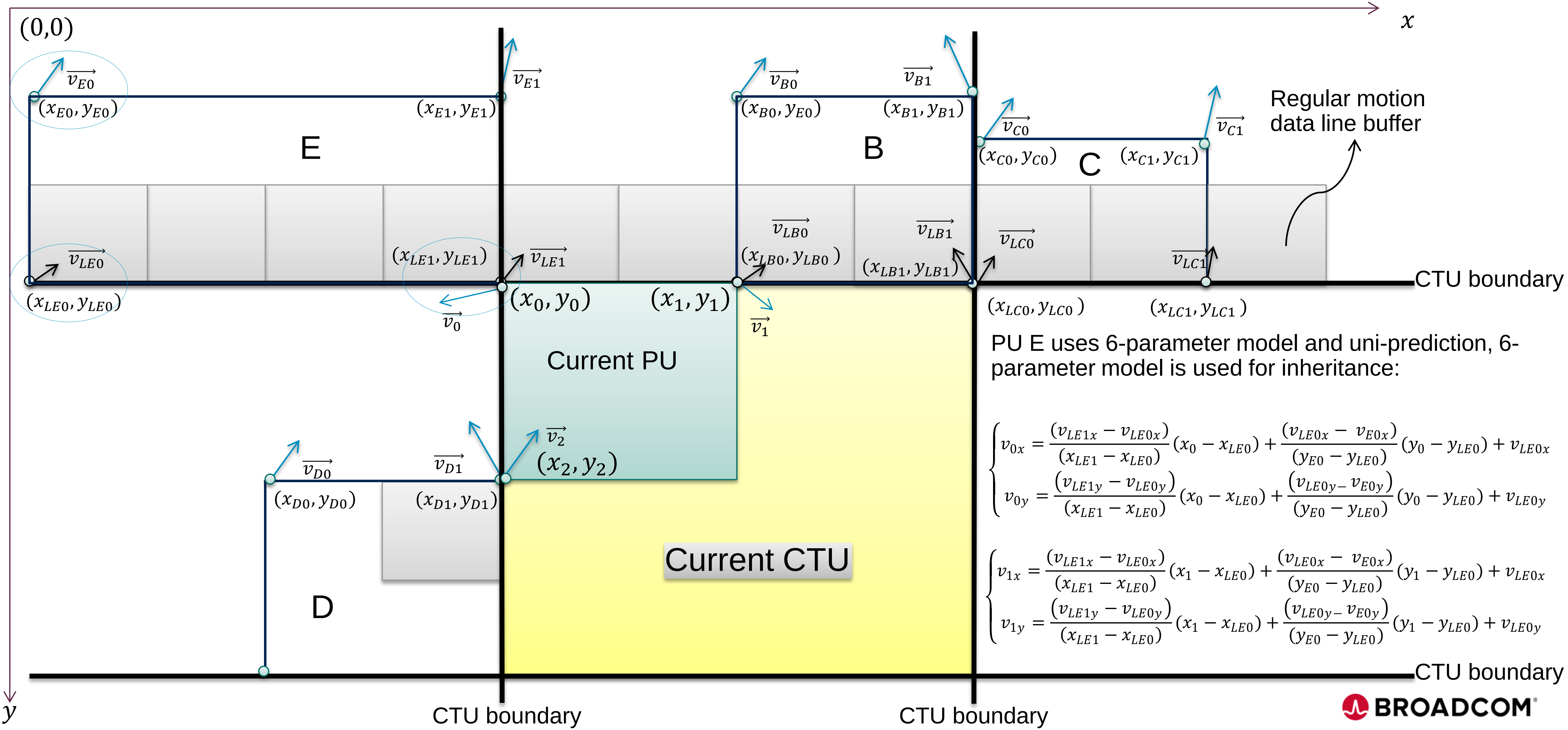
PU E uses 4- parameter model, or uses 6-parameter model, bi-predication and PU width = 8, 4-parameter model is used for inheritance:

$$\begin{cases} v_{0x} = \dfrac{(v_{LE1x} - v_{LE0x})}{(x_{LE1} - x_{LE0})}(x_0 - x_{LE0}) - \dfrac{(v_{LE1y} - v_{LE0y})}{(x_{LE1} - x_{LE0})}(y_0 - y_{LE0}) + v_{LE0x} \\ v_{0y} = \dfrac{(v_{LE1y} - v_{LE0y})}{(x_{LE1} - x_{LE0})}(x_0 - x_{LE0}) + \dfrac{(v_{LE1x} - v_{LE0x})}{(x_{LE1} - x_{LE0})}(y_0 - y_{LE0}) + v_{LE0y} \end{cases}$$

$$\begin{cases} v_{1x} = \dfrac{(v_{LE1x} - v_{LE0x})}{(x_{LE1} - x_{LE0})}(x_1 - x_{LE0}) - \dfrac{(v_{LE1y} - v_{LE0y})}{(x_{LE1} - x_{LE0})}(y_1 - y_{LE0}) + v_{LE0x} \\ v_{1y} = \dfrac{(v_{LE1y} - v_{LE0y})}{(x_{LE1} - x_{LE0})}(x_1 - x_{LE0}) + \dfrac{(v_{LE1x} - v_{LE0x})}{(x_{LE1} - x_{LE0})}(y_1 - y_{LE0}) + v_{LE0y} \end{cases}$$

BROADCOM®

Case 2 (PU E uses 6-parameter model and uni-prediction)

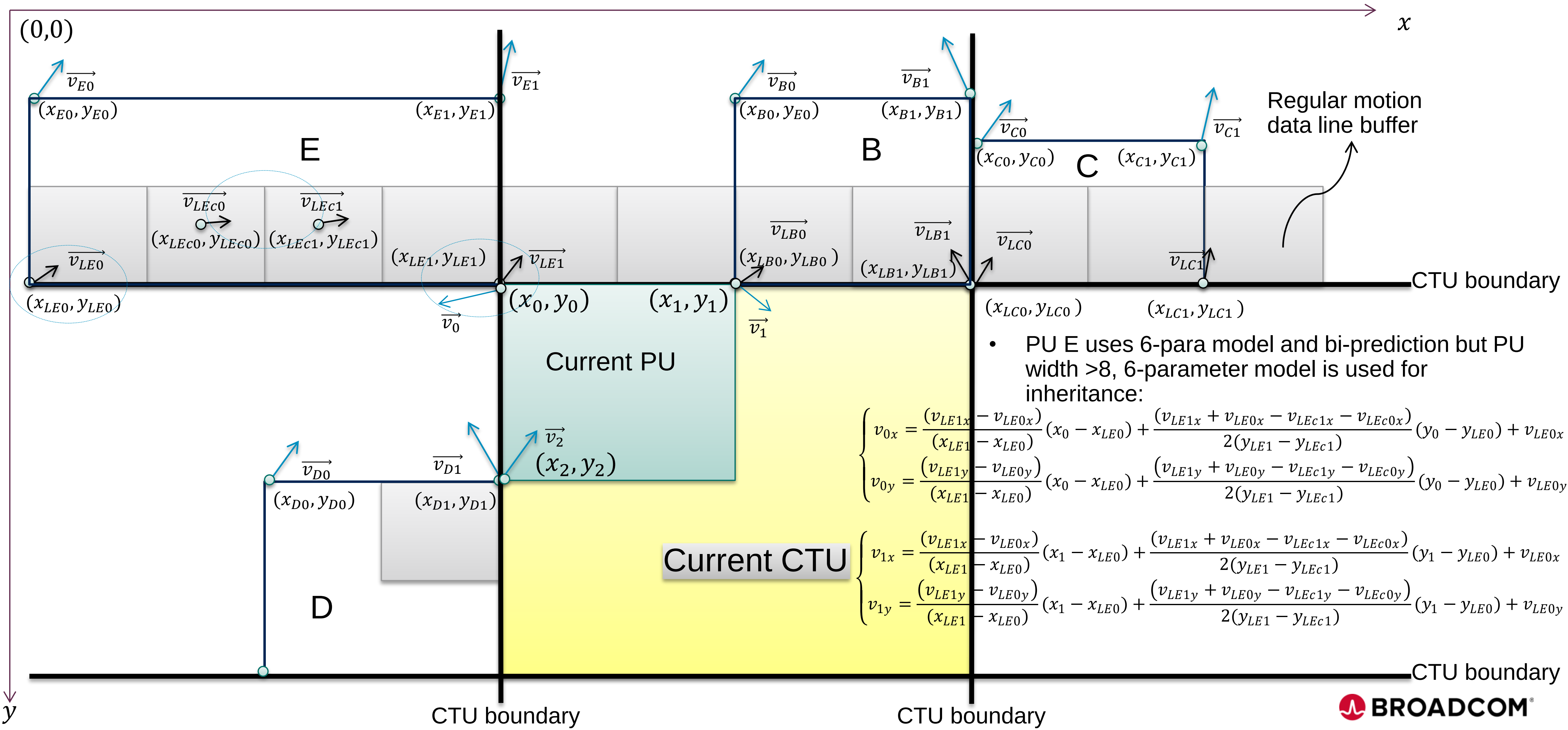


PU E uses 6-parameter model and uni-prediction, 6-parameter model is used for inheritance:

$$\begin{cases} v_{0x} = \dfrac{(v_{LE1x} - v_{LE0x})}{(x_{LE1} - x_{LE0})}(x_0 - x_{LE0}) + \dfrac{(v_{LE0x} - v_{E0x})}{(y_{E0} - y_{LE0})}(y_0 - y_{LE0}) + v_{LE0x} \\ v_{0y} = \dfrac{(v_{LE1y} - v_{LE0y})}{(x_{LE1} - x_{LE0})}(x_0 - x_{LE0}) + \dfrac{(v_{LE0y} - v_{E0y})}{(y_{E0} - y_{LE0})}(y_0 - y_{LE0}) + v_{LE0y} \end{cases}$$

$$\begin{cases} v_{1x} = \dfrac{(v_{LE1x} - v_{LE0x})}{(x_{LE1} - x_{LE0})}(x_1 - x_{LE0}) + \dfrac{(v_{LE0x} - v_{E0x})}{(y_{E0} - y_{LE0})}(y_1 - y_{LE0}) + v_{LE0x} \\ v_{1y} = \dfrac{(v_{LE1y} - v_{LE0y})}{(x_{LE1} - x_{LE0})}(x_1 - x_{LE0}) + \dfrac{(v_{LE0y} - v_{E0y})}{(y_{E0} - y_{LE0})}(y_1 - y_{LE0}) + v_{LE0y} \end{cases}$$

Case 3 (PU E uses Bi-prediction, but PU width >8)



- PU E uses 6-para model and bi-prediction but PU width >8, 6-parameter model is used for inheritance:

$$\begin{cases} v_{0x} = \frac{(v_{LE1x} - v_{LE0x})}{(x_{LE1} - x_{LE0})}(x_0 - x_{LE0}) + \frac{(v_{LE1x} + v_{LE0x} - v_{LEc1x} - v_{LEc0x})}{2(y_{LE1} - y_{LEc1})}(y_0 - y_{LE0}) + v_{LE0x} \\ v_{0y} = \frac{(v_{LE1y} - v_{LE0y})}{(x_{LE1} - x_{LE0})}(x_0 - x_{LE0}) + \frac{(v_{LE1y} + v_{LE0y} - v_{LEc1y} - v_{LEc0y})}{2(y_{LE1} - y_{LEc1})}(y_0 - y_{LE0}) + v_{LE0y} \end{cases}$$

$$\begin{cases} v_{1x} = \frac{(v_{LE1x} - v_{LE0x})}{(x_{LE1} - x_{LE0})}(x_1 - x_{LE0}) + \frac{(v_{LE1x} + v_{LE0x} - v_{LEc1x} - v_{LEc0x})}{2(y_{LE1} - y_{LEc1})}(y_1 - y_{LE0}) + v_{LE0x} \\ v_{1y} = \frac{(v_{LE1y} - v_{LE0y})}{(x_{LE1} - x_{LE0})}(x_1 - x_{LE0}) + \frac{(v_{LE1y} + v_{LE0y} - v_{LEc1y} - v_{LEc0y})}{2(y_{LE1} - y_{LEc1})}(y_1 - y_{LE0}) + v_{LE0y} \end{cases}$$

Memory Reduction Analysis for the Proposed Algorithm

- About 9 K bytes reduction for 4K video

Luma picture width = 4096		line buffer for merge/skip and AMVP mode		line buffer for affine mode in VTM2.0.1		line buffer for affine mode after simplification	
	Bits/unit	Number of units per 4 luma samples	list0 & list1 line buffer size (bytes)	Number of units per 8 luma samples	list0 & list1 line buffer size (bytes)	Number of units per 8 luma samples	list0 & list1 line buffer size (bytes)
sub-block MVs/CPMVs	32	2	8192	4	8192	0	0
Reference index	4	2	1024	2	512	0	0
PU width	4	0	0	1	256	1	256
PU height	4	0	0	1	256	0	0
Total (bytes)			**9216**		**9216**		**256**

Overview of Reference Algorithm

- For the affine motion data inheritance from a neighboring PU located at the top CTU boundary,
  - If the neighboring PU uses the 4-parameter model, the 4-parameter model using the bottom-left and bottom-right control point motion vectors (CPMVs) is applied to the inheritance.
  - Otherwise, if the neighboring PU uses the 6-parameter model and unidirectional prediction, the 6-parameter model using the bottom-left, bottom-right and top-left CPMVs is applied to the inheritance.
- In all the other cases, mathematically equivalent inheritance (to the current VTM2.0.1 method) is used.
- Only cuts half of line buffer for affine mode, as the top-left CPMVs still need to be stored in a separate line buffer

Experimental Results (VTM2.0.1 code base)

- Left – proposed algorithm (peak loss 0.11% in RA) and Right - reference algorithm (peak loss 0.07% in RA)

Left – proposed algorithm

	Random Access Main 10				
	Over VTM-2.0.1				
	Y	U	V	EncT	DecT
Class A1	-0.01%	0.04%	-0.08%	95%	91%
Class A2	0.08%	0.00%	-0.03%	91%	92%
Class B	0.02%	-0.02%	-0.03%	90%	91%
Class C	0.00%	0.11%	0.06%	94%	98%
Class E					
Overall	0.02%	0.03%	-0.01%	92%	93%
Class D	0.00%	0.02%	-0.07%	99%	97%
Class F (optional)	-0.01%	-0.04%	-0.01%	93%	90%

	Low delay B Main10				
	Over VTM-2.0.1				
	Y	U	V	EncT	DecT
Class A1					
Class A2					
Class B	0.03%	-0.11%	-0.30%	90%	99%
Class C	0.04%	-0.08%	0.04%	86%	89%
Class E	-0.06%	-0.12%	-0.51%	102%	101%
Overall	0.01%	-0.10%	-0.24%	91%	96%
Class D	0.00%	0.33%	0.23%	117%	114%
Class F (optional)	-0.04%	0.56%	-0.03%	101%	105%

	Low delay P Main10				
	Over VTM-2.0.1				
	Y	U	V	EncT	DecT
Class A1					
Class A2					
Class B	-0.01%	0.12%	0.12%	95%	99%
Class C	0.00%	-0.03%	0.09%	94%	96%
Class E	0.15%	-0.09%	0.13%	84%	83%
Overall	0.03%	0.02%	0.11%	92%	94%
Class D	0.10%	0.00%	-0.98%	98%	91%
Class F (optional)	-0.03%	-0.13%	-0.40%	88%	85%

Right – reference algorithm

	Random Access Main 10				
	Over VTM-2.0.1				
	Y	U	V	EncT	DecT
Class A1	-0.03%	0.14%	-0.04%	95%	93%
Class A2	0.02%	0.01%	-0.03%	95%	94%
Class B	0.02%	0.02%	-0.08%	95%	92%
Class C	0.01%	0.14%	0.08%	102%	100%
Class E					
Overall	0.01%	0.07%	-0.02%	97%	95%
Class D	0.01%	-0.01%	-0.04%	106%	108%
Class F (optional)	-0.01%	0.00%	0.00%	99%	97%

	Low delay B Main10				
	Over VTM-2.0.1				
	Y	U	V	EncT	DecT
Class A1					
Class A2					
Class B	0.01%	-0.02%	-0.15%	99%	104%
Class C	0.03%	0.05%	0.01%	86%	86%
Class E	-0.02%	0.01%	-0.34%	101%	103%
Overall	0.01%	0.01%	-0.15%	95%	97%
Class D	0.06%	0.43%	-0.40%	104%	106%
Class F (optional)	-0.04%	0.21%	-0.05%	104%	106%

	Low delay P Main10				
	Over VTM-2.0.1				
	Y	U	V	EncT	DecT
Class A1					
Class A2					
Class B	-0.01%	0.12%	0.12%	96%	99%
Class C	0.00%	-0.03%	0.09%	108%	114%
Class E	0.15%	-0.09%	0.13%	88%	85%
Overall	0.03%	0.02%	0.11%	98%	100%
Class D	0.10%	0.00%	-0.98%	92%	93%
Class F (optional)	-0.03%	-0.13%	-0.40%	101%	97%

Thanks to Huawei for cross-check

BROADCOM