

# **Joint Algorithm-Architecture Optimization of CABAC (JCTVC-E324/m19852)**

Vivienne Sze, Anantha P. Chandrakasan  
Massachusetts Institute of Technology



**Joint Collaborative Team on Video Coding (JCT-VC)  
of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11**  
5th Meeting: Geneva, CH, 16-23 March, 2011



# Algorithm and Architecture

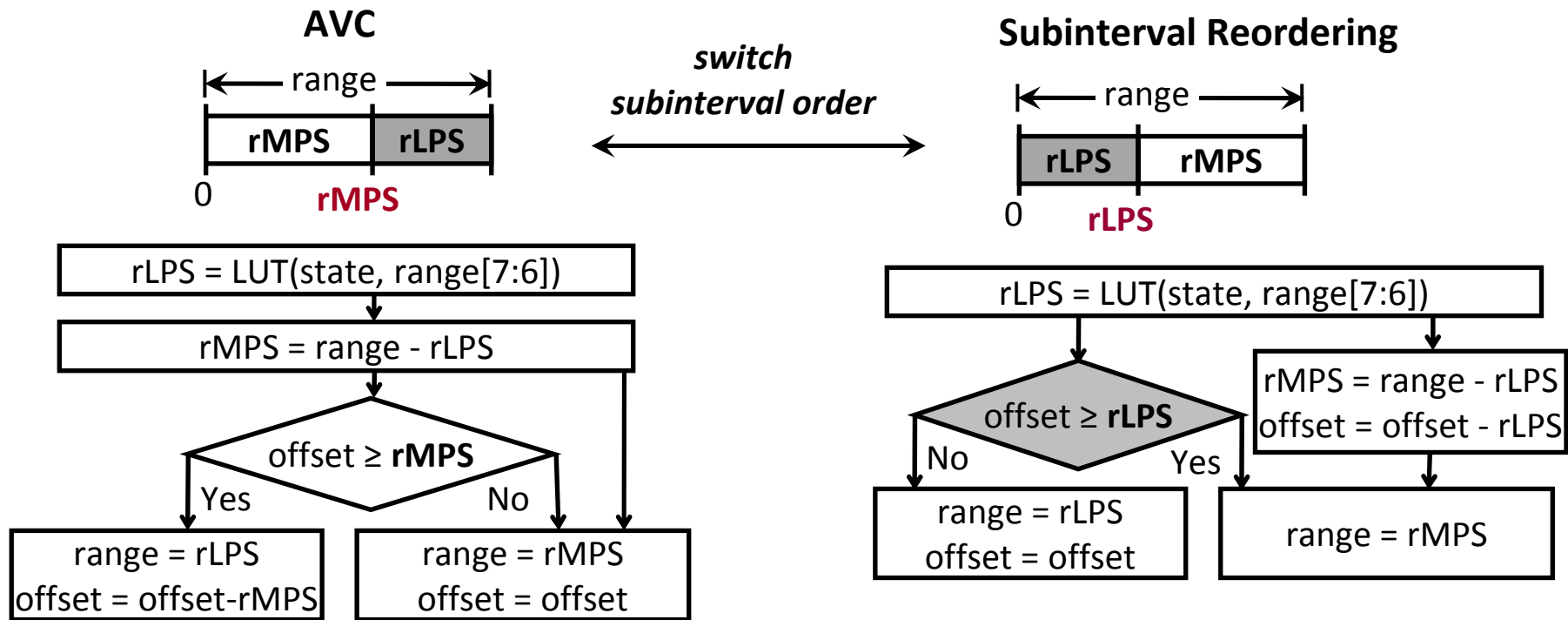
---

- Joint optimization of both the algorithm and architecture for high coding efficiency to
  - Increase processing speed
  - Reduce memory requirements
- Propose two modifications to CABAC
  1. Subinterval Reordering
  2. Modified MVD context selection



# Subinterval Reordering

- Placing rLPS at bottom of range enables parallel operations to increase speed
  - Subtract rLPS from range and offset (i.e. comparison) in parallel

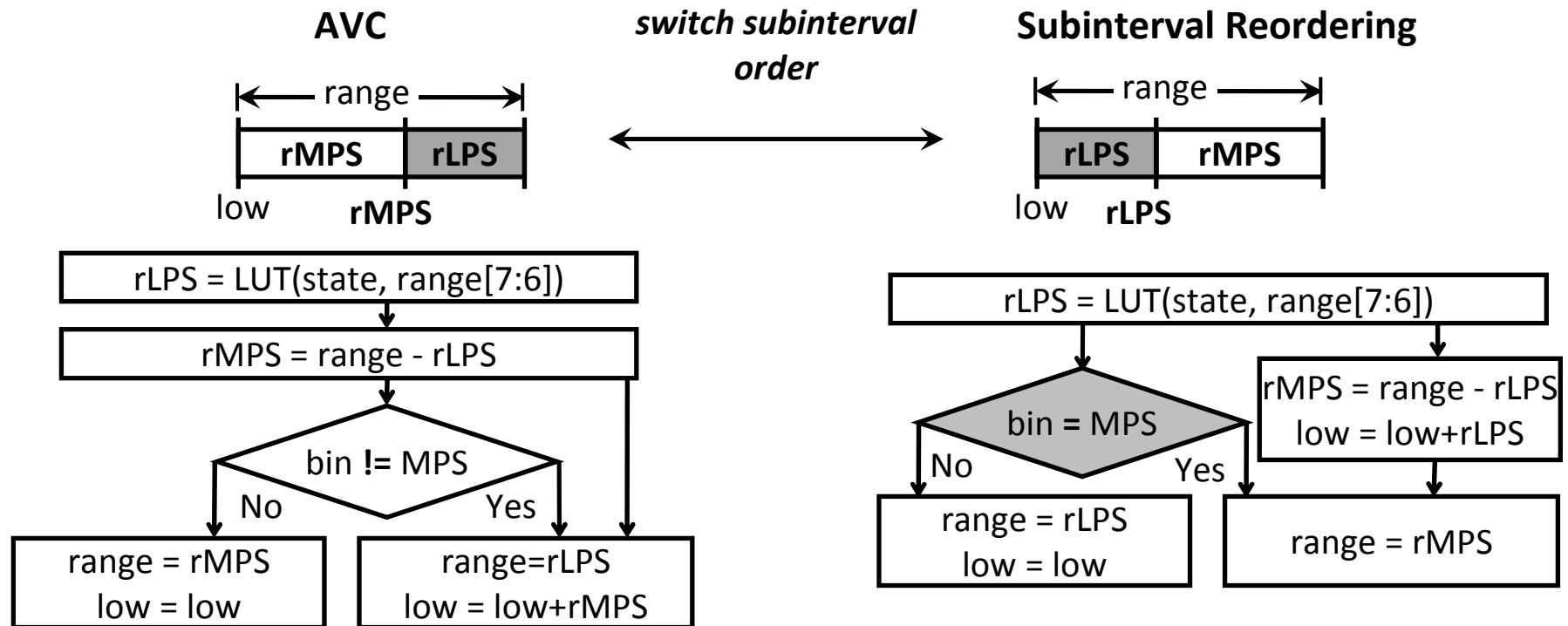


Reduce critical path delay by 14 to 22 %  
No impact on coding efficiency



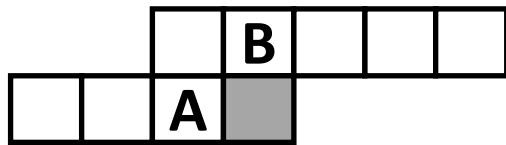
# Subinterval Reordering

- Similar benefits at encoder





# Modified MVD Context Selection



Context selection ( $\chi_{incr}$ ) dependent on A & B (4x4 blocks for *mvd*); as a result, last line buffer is required.

## AVC

### (1) SUM:

$$e(A,B) = |mvd(A)| + |mvd(B)|$$

### (2) THRESHOLD:

$$\chi_{incr} = \begin{cases} 0 \rightarrow \text{if } e(A,B) < 3 \\ 1 \rightarrow \text{if } 3 \leq e(A,B) \leq 32 \\ 2 \rightarrow \text{if } e(A,B) > 32 \end{cases}$$

## This work

### (1) THRESHOLD:

$$\text{threshA} = |mvd(A)| > 16$$

$$\text{threshB} = |mvd(B)| > 16$$

### (2) SUM:

$$\chi_{incr} = \text{threshA} + \text{threshB}$$

**If switch order, reduce from 6-bits to 1-bits stored per component**

Reduce overall CABAC Last Line Buffer Size by 50%  
Negligible impact on coding efficiency



# Experiment Results

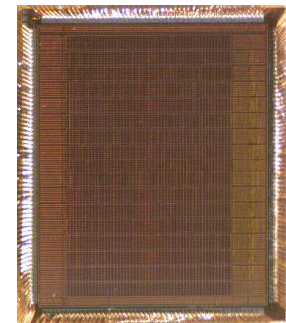
- HM-2.0 under common conditions
- Simulation platform is LSF equipped with Intel(R) Xeon(R) CPU X5570@2.93GHz 64 bits Linux machines
- Results cross-checked by HHI

## Coding efficiency impact of joint optimizations

	Intra	Random Access	Low Delay
Subinterval reorder & modified mvd context selection	0.0	-0.01	-0.03

## Optimizations Demonstrated in Hardware

- V. Sze, A. P. Chandrakasan, "A Highly Parallel and Scalable CABAC Decoder for Next-Generation Video Coding," IEEE International Conference on Solid-State Circuits (ISSCC), pp. 126-127, February 2011.



65-nm CMOS



# Summary

---

- Speed up arithmetic coding engine by 14-22%
- Reduce last line buffer size in CABAC
- Negligible impact on coding efficiency
- Recommend for adoption into HEVC test model
  - Draft text available in contribution