

JVET-E0119

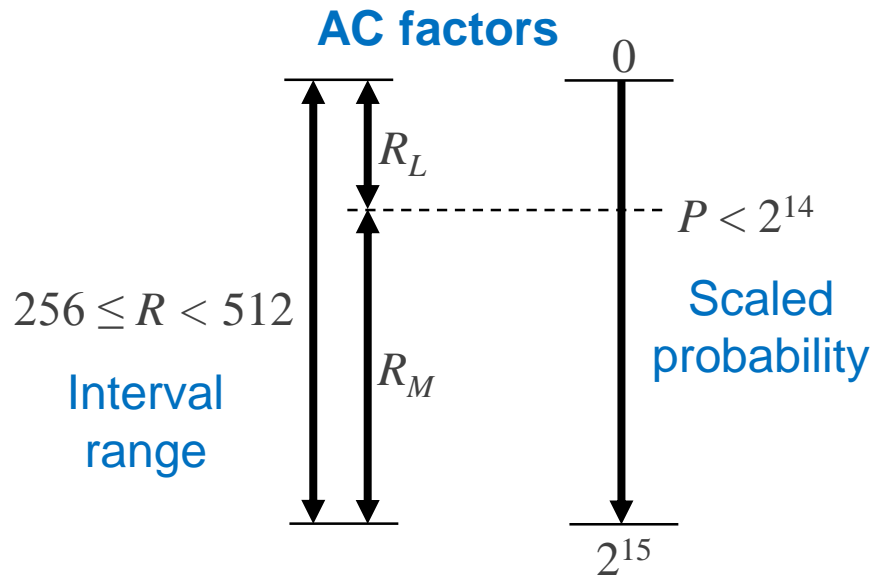
Binary arithmetic coding with small table or short multiplications

Amir Said, Ted Hsieh, Rajan Joshi, Vadim Seregin, Marta Karczewicz
Qualcomm Inc.

Arithmetic coding (AC) engine

- One of the most critical decoder components
 - Decoder capacity may be limited by CABAC's throughput
 - *Must* have very low complexity on software and hardware implementations
 - Tables have been used to replace multiplications
- Multiplication table size
 - HM: $64 \times 4 \times 8 = 2 \text{ Kbits}$
 - JEM: $512 \times 64 \times 9 = 295 \text{ Kbits}$
- Desirable features of future implementations
 - Implementable using *either multiplications or table look-up*
 - Total number of multiplication bits *must be very small*
 - Small tables that can reside in the L1 cache of cheap processors

AC implementation



AC multiplication

$$R_L \approx \left\lfloor \frac{R \times P}{2^{15}} \right\rfloor$$

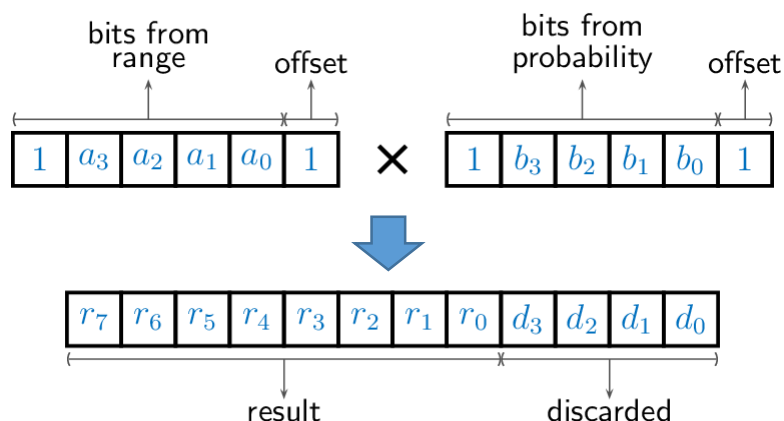
Current table

$$R_L = \mathbf{T}[\underbrace{P \gg 6}_{9 \text{ bits}}][\underbrace{(R \gg 2) \& 63}_{6 \text{ bits}}]$$

- Costs of N_p -bit \times N_r -bit multiplication
 - Number of gates is proportional to $N_p \times N_r$
 - Latency roughly proportional to $\max(N_p, N_r)$
- Simplification methods
 - Truncation to small number of bits
 - Reduced precision can severely degrade compression
 - Rounding
 - Trivial in hardware
 - Reduces speed of software-based table-look-up
 - Probability scaling
 - Replaces need for more operand bits with efficient bit shifts

Proposed method

Multiplication



New table

$$\mathbf{T}[m][n] = ((33 + 2m)(33 + 2n)) \gg 4$$

$$z = \lfloor \log_2(P) \rfloor$$

$$R_L = \underbrace{\mathbf{T}[(P \gg (z - 4))\&15]}_{4 \text{ bits}} [\underbrace{(R \gg 4)\&15}]_{4 \text{ bits}} \ll (13 - z)$$

- Normalize probability using bit shifts
 - Simple and efficient
 - Most-significant bits are always “1”
- New simplification
 - Add offset to low-precision operands
 - Equivalent to adding 1/2 to integer operands
 - Least-significant bits are always “1”
 - For a given accuracy (coding performance) provides
 - Smallest tables
 - Simplest hardware
- Proposed values
 - 4 bits for table indexes
 - 6 bits for multiplication (2 are always “1”)
 - 8 bits for results & table entries
- Multiplication table size = 2 Kbits (same as HM)

CTC Simulation Results

	All Intra Main10				
	Over JEM-4.1 (parallel)				
	Y	U	V	EncT	DecT
Class A1	0.01%	0.01%	0.01%	101%	101%
Class A2	0.00%	0.00%	0.00%	101%	101%
Class B	0.00%	0.00%	0.00%	100%	101%
Class C	0.00%	0.00%	0.00%	100%	101%
Class D	0.00%	0.00%	0.00%	100%	101%
Class E	0.00%	0.00%	0.00%	100%	100%
Overall	0.00%	0.00%	0.00%	100%	101%
Class F (optional)	0.00%	0.00%	0.00%	100%	101%

	Random Access Main 10				
	Over JEM-4.1 (parallel)				
	Y	U	V	EncT	DecT
Class A1	#VALUE!	#VALUE!	#VALUE!	#DIV/0!	#DIV/0!
Class A2	#VALUE!	#VALUE!	#VALUE!	#DIV/0!	#DIV/0!
Class B	-0.01%	-0.01%	-0.01%	100%	100%
Class C	0.01%	0.01%	0.01%	100%	100%
Class D	0.00%	0.00%	0.00%	100%	100%
Class E					
Overall (Ref)	#VALUE!	#VALUE!	#VALUE!	#DIV/0!	#DIV/0!
Class F (optional)	-0.01%	-0.01%	-0.01%	100%	100%