INTERNATIONAL ORGANISATION FOR STANDARDISATION
ORGANISATION INTERNATIONALE DE NORMALISATION
ISO/IEC JTC1/SC29/WG11
CODING OF MOVING PICTURES AND AUDIO

ISO/IEC JCTC1/SC29/WG11 MPEG/m52514
January 2020, Brussels, Belgium

Source: Apple Inc.
Status: Input document
Title: CE13.10 report on chunked entropy streams
Author(s): David Flynn | *davidflynn@apple.com*
Khaled Mammou | *kmammou@apple.com*

Abstract

This document reports on using a chunked interleaving of an arithmetically coded sub stream and a bypass sub stream as part of core experiment 13.10.

Introduction

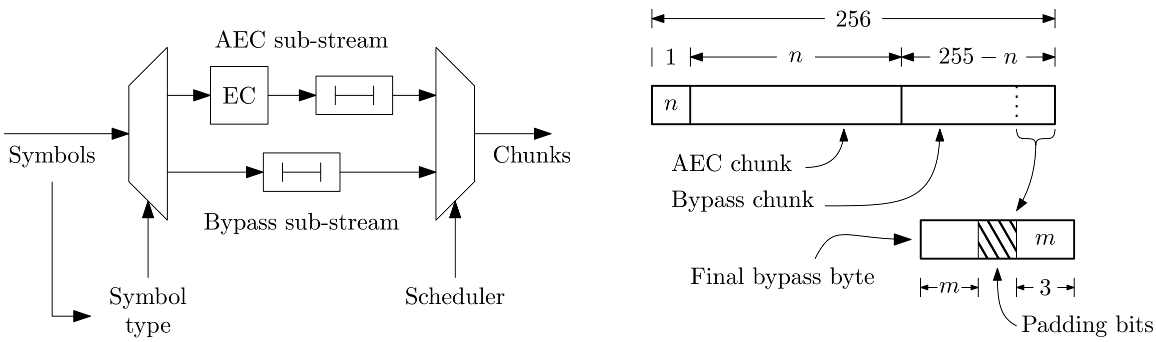
The mandate of the core experiment is to evaluate the use of a chunk-based interleaving method [1] for representing sub-streams of arithmetic and bypass coded data to —

- Examine the coding efficiency of the proposed method.
- Examine the buffering requirements for a decoder.

Overview

The introduction of the separate coding of a sub stream consisting of bypass symbols was previously studied in [2, 3].

The proposed method is an alternative approach based on a chunk-interleaved representation of the two sub-streams. It aims to balance the benefit of not arithmetically coding bypass bins, the ability to transmit and receive the bitstream as a whole in the forward order, with the chunk signalling overhead.



(a) Splitting of sub-streams

(b) A chunk

Figure 1 – Construction of a multiplexed sub-stream chunk

Each 256 byte chunk is formed of (shown diagrammatically in fig. 1):

- a one byte header that indicates the number of arithmetically coded bytes present in the chunk, n,
- n bytes of arithmetically coded data, and

- $255 - n$ bytes of (non arithmetically coded) bypass data.

Special handling of the last bypass coded byte in a chunk allows an encoder to flush incomplete chunks in order to permit low latency coding and decoding. The last bypass byte, if present, contains a three bit value indicating the number of bits that have been flushed and should be discarded from the bypass sub stream.

In the original contribution, the flushing aspect was optional. However, during the presentation and discussion in the 128th meeting it was recommended to remove the optionality. This CE study follows this request and studies only the version with explicit flushing support.

Implementation

An implementation has been provided in the `mpeg128/ce13.10/chunked-entropy-stream` branch of the CE repository. It behaves as follows:

- on each byte output by the arithmetic encoder an attempt is made to write it to the current chunk.
- bits from the bypass stream are written by attempting to reserve a byte from the current chunk and then filling the reserved byte with subsequent bits. The fullness measure takes into account the need to ensure there is sufficient space for the padding length indicator at the end of the chunk.

If a byte cannot be allocated within or reserved from the current chunk, the chunk is flushed and a new chunk started.

The decoder operates using two pointers that each independently seek though the chunks for their respective data. However, more optimal implementations can single pointer.

Results

To evaluate the performance of the method using TMC13v8 [4], two anchors are defined based upon the common test conditions [5], one for bitwise occupancy coding, the other for bytewise occupancy coding. Both anchors deviate from the common test conditions by enabling the bypass substream (since the test subject modifies the bypass substream). Summary results are presented in Tables 1 to 5.

Table 1 – Performance using bitwise octree geometry coding and LoD attribute coding

		BPP Ratio [%]			BD-Rate [Δ%]						Avg. of ratio maxrssk [%]		Ratio of avg. runtime [%]	
Condition	Class	Geometry	Colour	Refl	D1	D2	Y	Cb	Cr	R	Encoder	Decoder	Encoder	Decoder
C1_ai	cat1-A						0.5!	0.5!	0.4!		100	100	101	100
C1_ai	cat3-fused						0.5	0.5	0.5	0.4	100	100	103	102
C1_ai	cat3-frame									0.9!	100	100	106	107
C1_ai	overall						0.5!	0.5!	0.4!	0.7!	100	100	103	103
C2_ai	cat1-A				0.5	0.4	0.5	0.5	0.5		100	100	98	99
C2_ai	cat1-B				0.6	0.6					100	100	102	
C2_ai	cat3-fused				0.7	0.7	0.5	0.5	0.5	0.4	100	100	106	106
C2_ai	cat3-frame				0.7	0.7				0.7	100	100	101	101
C2_ai	overall				0.6	0.6	0.5!	0.5!	0.5!	0.7	100	100	100	
CW_ai	cat1-A	100.6	100.6								100	100	103	98
CW_ai	cat1-B	100.7									100	100	98	98
CW_ai	cat3-fused	100.7	100.7	100.7							100	100	99	105
CW_ai	cat3-frame	100.6		100.7							100	100	99	99
CW_ai	overall	100.7	100.6!	100.7							100	100	100	98
CY_ai	cat1-A						0.4!	0.4!	0.4!		100	100	101	103
CY_ai	cat3-fused						0.5	0.5	0.5	0.5	100	100	103	108
CY_ai	cat3-frame									0.5	100	100	111	111
CY_ai	overall						0.4!	0.4!	0.4!	0.5	100	100	103	105

NOTE — Condition CY metrics reported using Hausdorff PSNR.

Table 2 – Performance using bytewise octree geometry coding and LoD attribute coding

Condition	Class	BPP Ratio [%] Geometry	Colour	Refl	BD-Rate [Δ%] D1	D2	Y	Cb	Cr	R	Avg. of ratio maxrssk [%] Encoder	Decoder	Ratio of avg. runtime [%] Encoder	Decoder
C1_ai	cat1-A						0.5!	0.5!	0.4!		100	100	100	100
C1_ai	cat3-fused						0.5	0.5	0.5	0.4	100	100	103	103
C1_ai	cat3-frame									0.5	100	100	99	98
C1_ai	overall						0.5!	0.5!	0.5!	0.5	100	100	99	99
C2_ai	cat1-A				0.9!	0.8!	0.5!	0.5!	0.5!		100	100	104	101
C2_ai	cat1-B				0.6	0.6					100	100	99	
C2_ai	cat3-fused				0.6	0.6	0.5	0.5	0.5	0.4	100	100	94	97
C2_ai	cat3-frame				0.7	0.7				0.7	100	100	102	102
C2_ai	overall				0.7!	0.7!	0.5!	0.5!	0.5!	0.7	100	100	101	
CW_ai	cat1-A	100.6	100.6								100	100	98	99
CW_ai	cat1-B	100.6									100	100	102	112
CW_ai	cat3-fused	100.6	100.7	100.7							100	100	102	99
CW_ai	cat3-frame	100.6		100.7							100	100	106	104
CW_ai	overall	100.6	100.6!	100.7							100	100	101	105
CY_ai	cat1-A						0.4	0.4	0.4		100	100	97	99
CY_ai	cat3-fused						0.5	0.5	0.5	0.5	100	100	104	102
CY_ai	cat3-frame									0.5	100	100	101	101
CY_ai	overall						0.4	0.4	0.4	0.5	100	100	98	100

NOTE — Condition CY metrics reported using Hausdorff PSNR.

Table 3 – Performance using bitwise octree geometry coding and RAHT attribute coding

Condition	Class	BPP Ratio [%] Geometry	Colour	Refl	BD-Rate [Δ%] D1	D2	Y	Cb	Cr	R	Avg. of ratio maxrssk [%] Encoder	Decoder	Ratio of avg. runtime [%] Encoder	Decoder
C1_ai	cat1-A						0.4	0.4	0.4		100	100	99	100
C1_ai	cat1-B						0.4!	0.4!	0.4!		100	100	102	100
C1_ai	cat3-fused						0.4	0.5	0.5	0.4	100	100	104	101
C1_ai	cat3-frame									0.6	100	100	100	100
C1_ai	overall						0.4!	0.4!	0.4!	0.5	100	100	101	100
C2_ai	cat1-A				0.5!	0.5!	0.5!	0.5!	0.5!		100	100	100	102
C2_ai	cat1-B				0.6	0.6	0.5	0.5	0.5		100	100	101	96
C2_ai	cat3-fused				0.7	0.7	0.4	0.4	0.4	0.4	100	100	100	98
C2_ai	cat3-frame				0.6!	0.6!				1.0!	100	100	103	101
C2_ai	overall				0.6!	0.6!	0.5!	0.5!	0.5!	0.8!	100	100	101	99

Table 4 – Performance using bitwise octree geometry coding, trisoup, and LoD attribute coding

Condition	Class	BPP Ratio [%] Geometry	Colour	Refl	BD-Rate [Δ%] D1	D2	Y	Cb	Cr	R	Avg. of ratio maxrssk [%] Encoder	Decoder	Ratio of avg. runtime [%] Encoder	Decoder
C2_ai	cat1-A				0.4!	0.4!	0.6!	0.5!	0.7!		100	100	99	99

NOTE — Condition CY metrics reported using Hausdorff PSNR.

Table 5 – Performance using bitwise octree geometry coding, trisoup, and RAHT attribute coding

Condition	Class	BPP Ratio [%] Geometry	Colour	Refl	BD-Rate [Δ%] D1	D2	Y	Cb	Cr	R	Avg. of ratio maxrssk [%] Encoder	Decoder	Ratio of avg. runtime [%] Encoder	Decoder
C2_ai	cat1-A				0.4!	0.4!	0.4!	0.4!	0.4!		100	100	99	100
C2_ai	cat1-B				0.4!	1.4!	0.3!	0.3!	0.3!		100	100	99	100
C2_ai	overall				0.4!	1.0!	0.3!	0.4!	0.4!		100	100	99	100

References

[1] D. Flynn, A. Tourpis, and K. Mammou, “[G-PCC][New proposal] G-PCC low latency bypass bin coding,” ISO/IEC JTC1/SC29/WG11, 128th meeting, Geneva, Tech. Rep. m51024, Oct. 2019.

[2] 3DG, “CE 13.10 on entropy coding evaluation,” ISO/IEC JTC1/SC29/WG11, 126th meeting, Geneva, Tech. Rep. w18483, Mar. 2019.

[3] D. Flynn and S. Lasserre, “G-PCC CE13.10 report on bypass bin coding,” ISO/IEC JTC1/SC29/WG11, 127th meeting, Gothenburg, Tech. Rep. m49379, Jul. 2019.

[4] 3DG, “G-PCC Test Model v8,” ISO/IEC JTC1/SC29/WG11, 128th meeting, Geneva, Tech. Rep. w18882, Oct. 2019.

[5] ——, “Common Test Conditions for PCC,” ISO/IEC JTC1/SC29/WG11, 128th meeting, Geneva, Tech. Rep. w18883, Oct. 2019.