Thor update

High Efficiency, Moderate Complexity Video Codec using only RF IPR

(https://datatracker.ietf.org/ipr/2636/)

draft-fuldseth-netvc-thor-03

Steinar Midtskogen (Cisco)

IETF 98 – Chicago, US – March 2017

Changes in Thor since IETF97

- Support for monochrome video added
- Support for video with 4:2:2 chroma sampling
  - encoded as 4:4:4 internally
    - very simple to implement and the code remains clean and simple
    - suboptimal compression, 4:2:2 a corner case, complexity avoided for something few will use
- The constrained low-pass filter (CLPF) improved
  - Gives ~0.4% BDR gain
- Misc fixes (hbd CFL, portability, code sync with AV1)

Improved CLPF

- Increase from 6 to 8 taps:

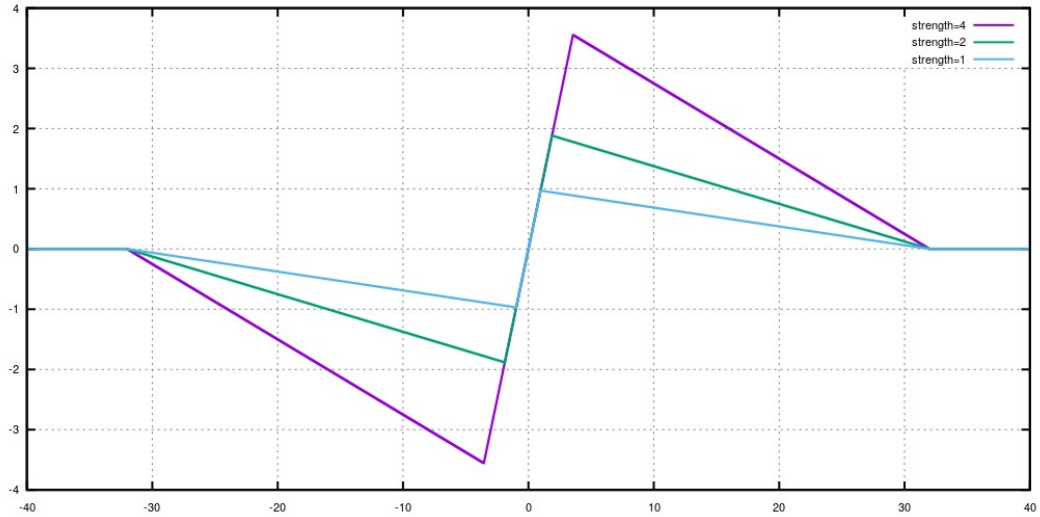
		A		
		B		
C	D	X	E	F
		G		
		H		

		1		
		3		
1	3		3	1
		3		
		1		

- A ramp-down added to the clip function
  - Most of the gain comes from this new clip function

Improved CLPF

- New clip function:



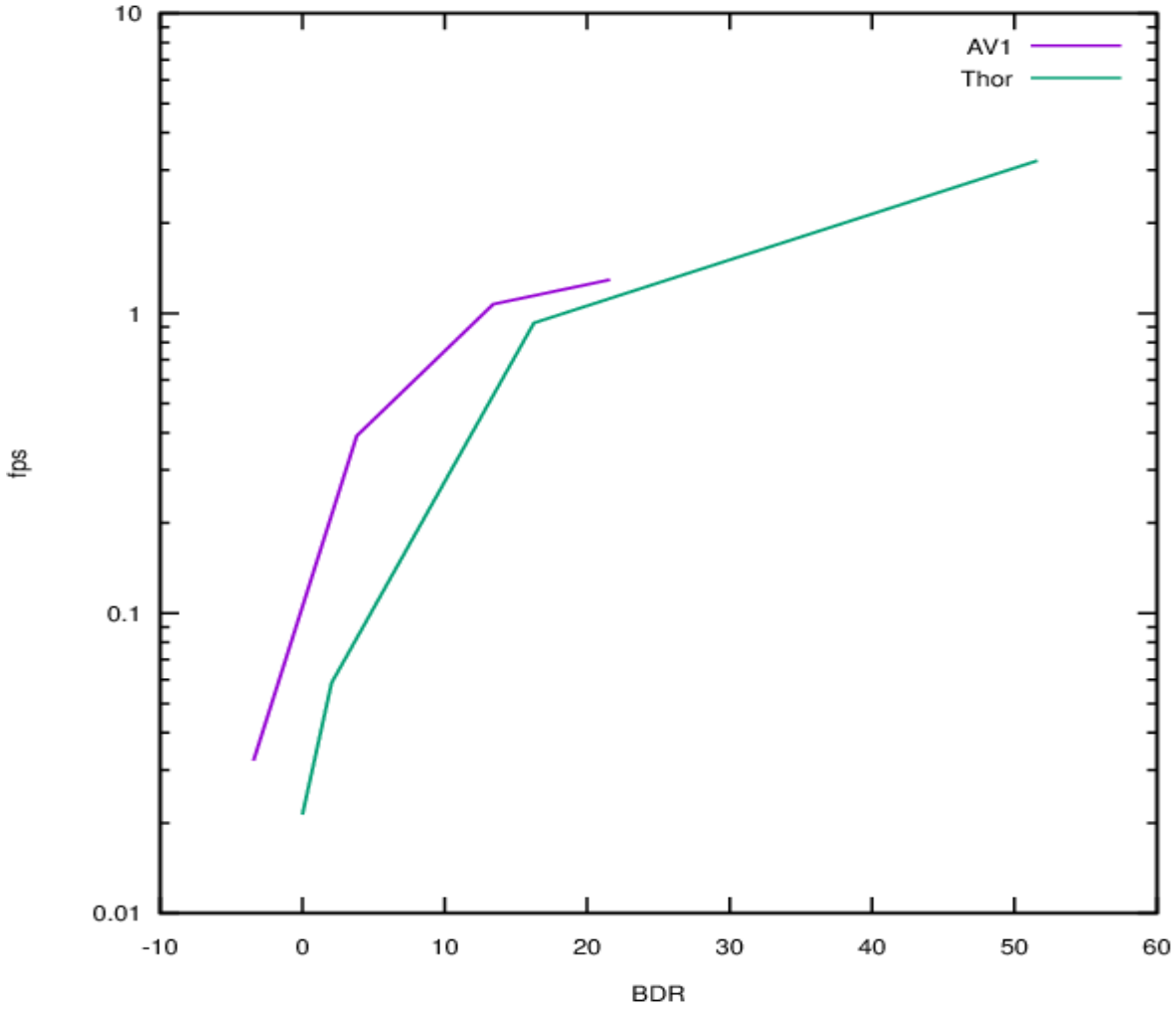
- Simple C code, SIMD friendly and easy to implement:

```
int constrain(int x, int s, unsigned int bitdepth) {
  return (x < 0 ? -1 : 1) *
    max(0, abs(x) - max(0, abs(x) - s + (abs(x) >> (bitdepth - 3 - log2(s)))));
}
int clpf_pixel(int X, int A, int B, int C, int D, int E, int F, int G, int H,
               int s, unsigned int bitdepth) {
  int delta = 1 * constrain(A - X, s, bitdepth) + 3 * constrain(B - X, s, bitdepth) +
              1 * constrain(C - X, s, bitdepth) + 3 * constrain(D - X, s, bitdepth) +
              3 * constrain(E - X, s, bitdepth) + 1 * constrain(F - X, s, bitdepth) +
              3 * constrain(G - X, s, bitdepth) + 1 * constrain(H - X, s, bitdepth);
  return X + (8 + delta - (delta < 0)) >> 4;  // Rounding, assumes arithmetic shift
```

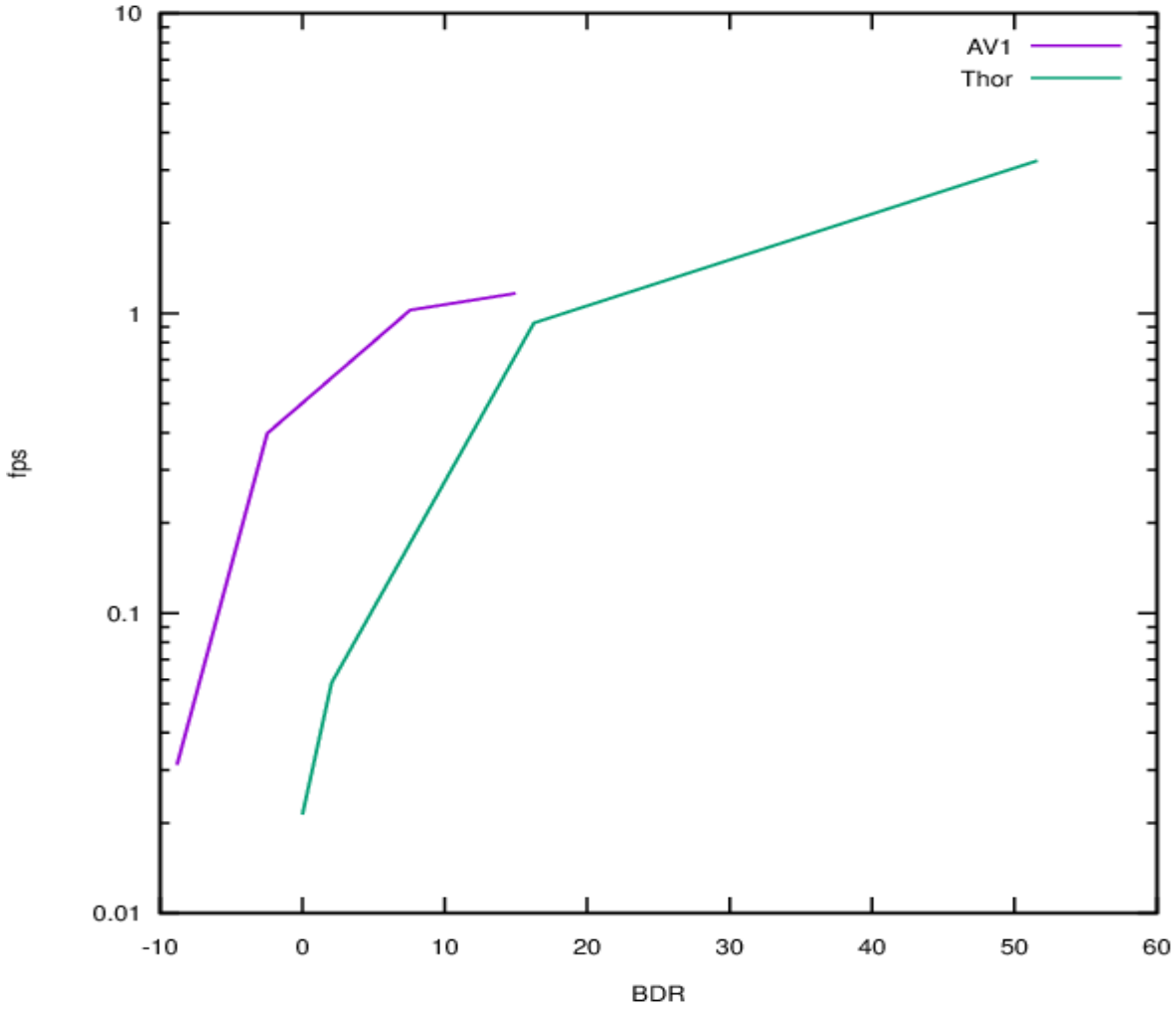
Thor compared to AV1

- Much activity in AV1 recently, not so much in Thor
- Codec performance now measured using AWCY
- AV1 now generally performs better than Thor
  - Many AV1 improvements and new tools
  - Thor may still be slightly better at low delay videoconferencing (meeting rooms and “talking heads”)
  - AV1 is much better than Thor at screen content
  - AV1 seems to have a 3-4x speed advantage over Thor on the AWCY servers compared to our servers, unknown reason
  - Error resilience has a significant cost
- AV1 is a moving target and will probably improve further by ~10% in the coming months

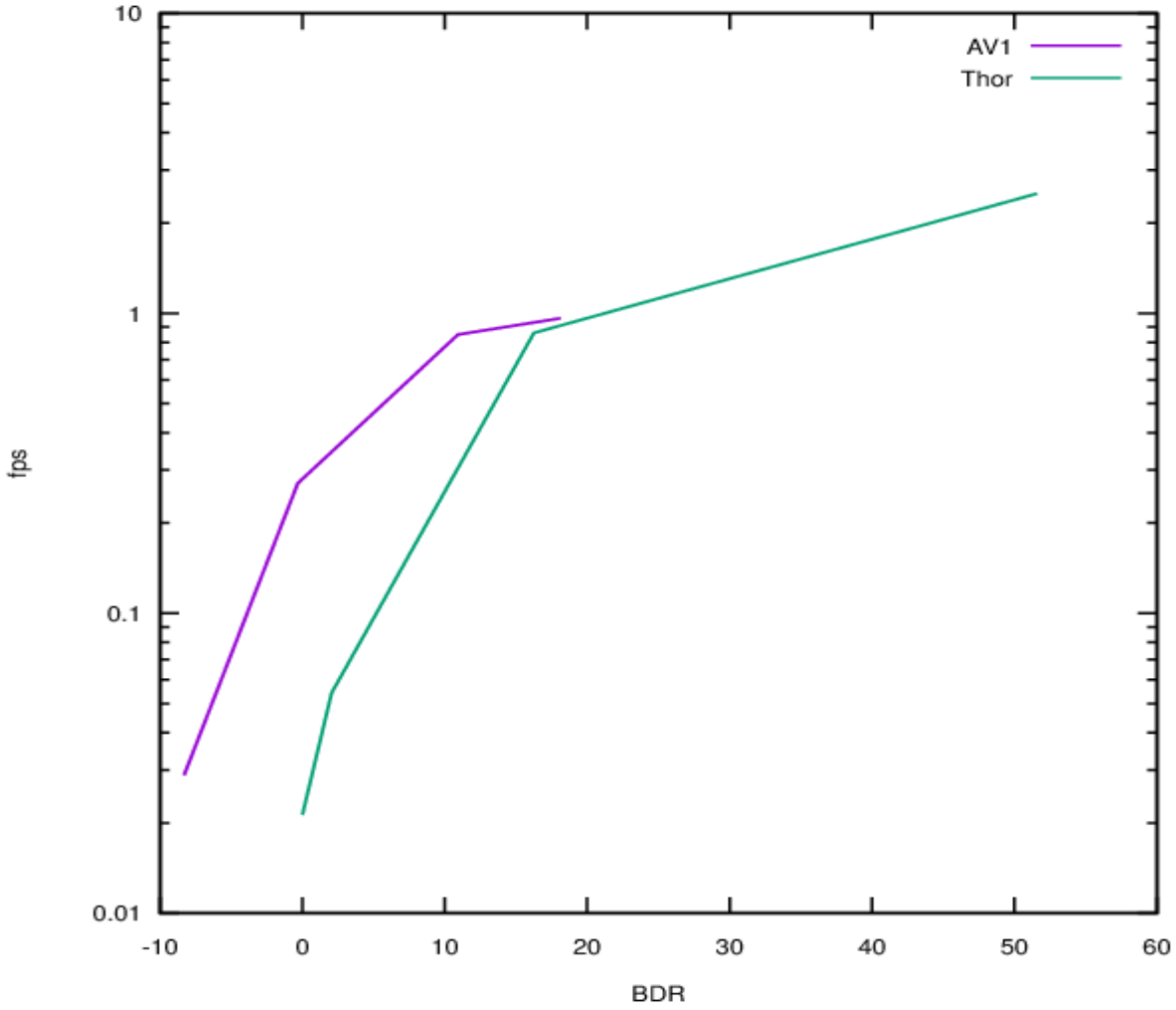
Low delay, error resilient



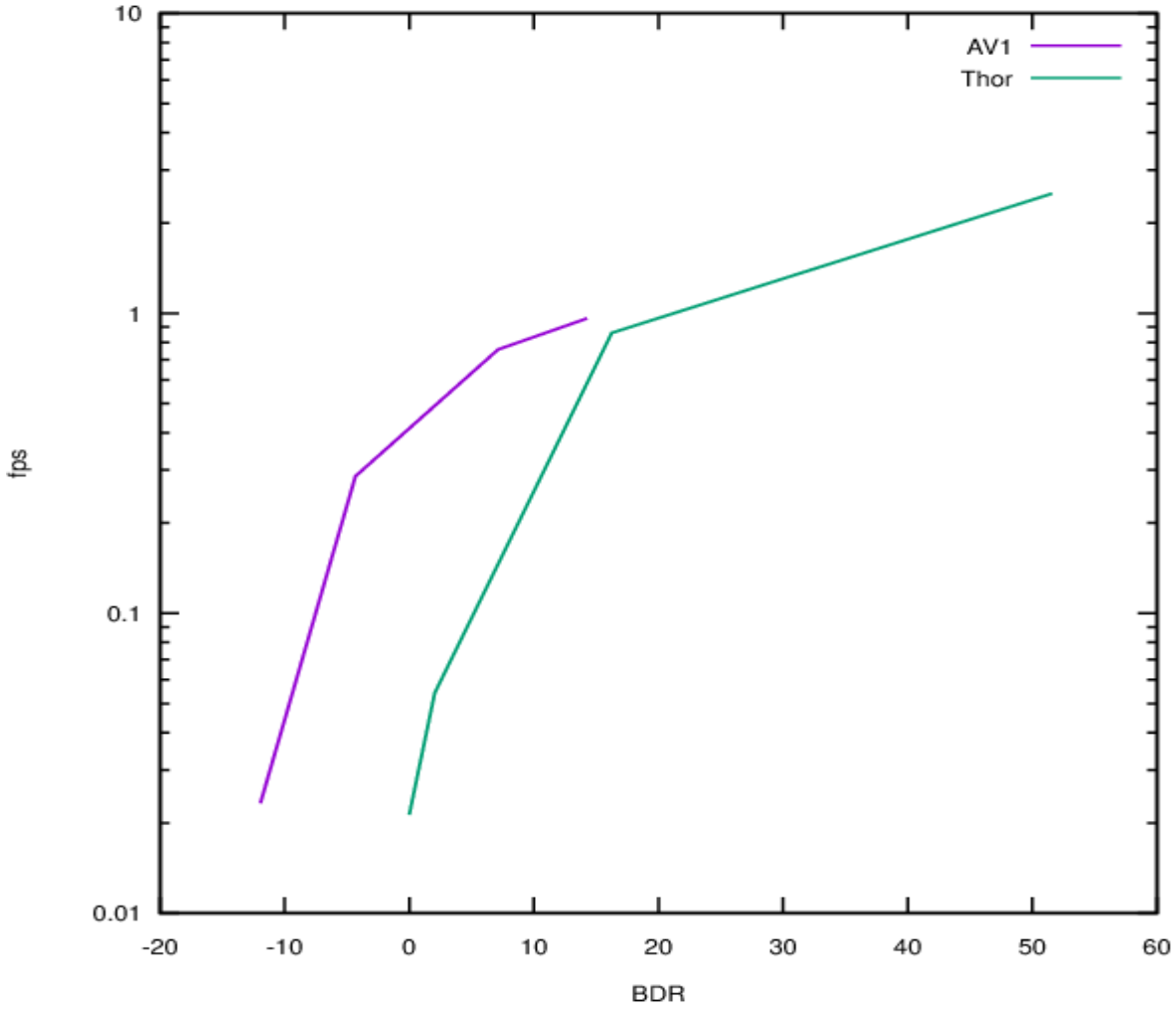
Low delay, not error resilient



High delay, error resilient



High delay, not error resilient



Differing results

- The results depend a lot on the sequence test
- Change from Thor to AV1 at similar complexity for some sequences as reported by AWCY:

kirland360p_60f.y4m	20.18	18.77	36.50	18.26	13.64	19.14	19.99	11.86	18.85	24.84
life_1080p30_60f.y4m	-17.24	-19.94	-20.82	-22.77	-38.85	-38.43	-17.53	-38.56	-38.29	-20.16
niklas360p_60f.y4m	-3.93	-7.02	-1.43	-12.31	-22.83	-23.46	-4.07	-22.83	-23.75	-6.64
red_kayak_360p_60f.y4m	-8.88	-9.85	-13.08	-12.62	-17.08	-19.17	-10.40	-18.36	-20.08	-13.59
rush_hour_1080p25_60f.y4m	13.00	10.98	12.67	4.87	-8.47	-15.50	12.71	-8.81	-15.72	11.67
shields_640x360_60f.y4m	-10.65	-10.27	-8.30	-8.80	-16.18	-19.34	-9.71	-16.84	-18.84	-8.83
speed_bag_640x360_60f.y4m	-16.05	-11.34	-6.60	-22.36	-32.54	-30.55	-17.43	-33.44	-32.03	-10.09
thaloundeskmtg360p_60f.y4m	-43.41	-38.83	-38.29	-43.92	-49.93	-48.02	-38.10	-48.61	-47.31	-38.01
touchdown_pass_1080p_60f.y4m	-0.37	-4.57	-5.88	-2.04	-24.73	-16.97	-2.63	-25.82	-18.21	-4.85
vidyo1_720p_60fps_60f.y4m	4.62	0.94	12.39	-2.04	-15.03	-21.51	4.51	-15.57	-21.50	7.87
vidyo4_720p_60fps_60f.y4m	5.57	2.55	8.21	-5.66	-27.02	-21.97	5.09	-26.85	-21.72	5.96
wikipedia_420.y4m	-79.82	-79.88	-80.09	-79.69	-79.97	-80.64	-79.60	-79.84	-80.57	-80.22

The road ahead (thinking loudly)

- Add arithmetic coding
  - Use the Daala entropy coder?
  - The entropy coder is in the core of a codec, so this should perhaps rather be regarded as a merge of Thor and Daala
- Merge CLPF and Daala dering (AV1 CDEF)
- Since the above tools have been adopted in AV1, this path would take the codec towards (a subset of) AV1
  - Other tools from Thor already adopted in AV1:
    - 7 bit interpolation filters
    - quantisation matrices
    - delta-q