

HEVC Performance and Complexity for 4K Video

Benjamin Bross*, Valeri George*, Mauricio Alvarez-Mesa*[†], Tobias Mayer*, Chi Ching Chi*[†],
Jens Brandenburg* Thomas Schierl*, Detlev Marpe*, and Ben Juurlink[†]

*Image Processing Department, Fraunhofer HHI, 10587 Berlin, Germany

[†]Embedded Systems Architecture Group, Technical University of Berlin, 10587 Berlin, Germany

Abstract—The recently finalized High-Efficiency Video Coding (HEVC) standard was jointly developed by the ITU-T Video Coding Experts Group (VCEG) and the ISO/IEC Moving Picture Experts Group (MPEG) to improve the compression performance of current video coding standards by 50%. Especially when it comes to transmit high resolution video like 4K over the internet or in broadcast, the 50% bitrate reduction is essential. This paper shows that real-time decoding of 4K video with a frame-level parallel decoding approach using four desktop CPU cores is feasible.

I. INTRODUCTION

In January 2013 and ten years after the widely-used H.264/MPEG4-AVC video coding standard [1] was published, the first version of the HEVC standard was finalized by ITU-T consent and issued as ISO/IEC Final Draft International Standard (FDIS) [2]. A design overview of the new HEVC standard can be found in [3]. The coding efficiency of HEVC was analyzed in [4] and compared with previous video coding standards like H.264/MPEG4-AVC and H.262/MPEG2-Video. Bitrate reductions of 50% for the same subjective quality compared to H.264/MPEG4-AVC are reported. Since this coding efficiency gain comes along with increased complexity, the complexity aspects of HEVC en-/decoding have been studied in [5] and [6] and en-/decoding times for HD (1920×1080) video sequences are reported. One of the targeted applications of HEVC is coding of ultra-high resolution video and hence, this paper reviews and reports results for real-time HEVC decoding of 4K (3840×2160) video sequences.

First approaches to enable real-time decoding of HEVC coded 4K video sequences have been analyzed and presented in [7], [8], [9], and [10]. In these studies, the HEVC test model (HM) reference software decoder code was optimized and modified to support multithreading. The first analysis uses multi-threading in combination with entropy slices in version 3.1 of the HM software [7]. Entropy slices are not part of the final standard but with Wavefront Parallel Processing (WPP), similar multi-threaded decoding as with entropy slices can be achieved. A slightly modified version of WPP, called Overlapped Wavefront (OWF) and Tiles have been studied in [8] and [9] based on HM 4.1. The most recent publication shows results for OWF based on HM 8.0 and further reports speedup due to the use of Single Instruction Multiple Data (SIMD) code optimizations [10]. The 4K (3840×2160) sequences used in these publications are from the Sveriges Television (SVT) High Definition Multi Format Test Set. Although WPP and Tiles allow low delay parallel decoding, a special indication

of these techniques in the bitstream is required.

II. REAL-TIME HEVC DECODING OF 4K VIDEO

In order to provide the required speedup for 4K decoding using parallel processing without putting constraints on the bitstream, e.g. having WPP or Tiles enabled, a frame-level parallel processing approach has been chosen for this paper. For the initial version of this approach presented here, each frame to be processed in parallel is assigned a worker thread. Therefore, the number of worker threads controls the number of frames to be processed in parallel.

The frame-level parallelism has been integrated in a from scratch HEVC decoder implementation developed at Fraunhofer HHI and results are provided for all sequences from the 4K (3840×2160) 50Hz UHD-1 test set provided by the European Broadcast Union (EBU) [11]. These have been encoded with version 10 of the HM reference software (HM10) [12] using the Intra Main, Intra High Efficiency 10 bit (Main 10), Random Access and Random Access High Efficiency 10 bit (Main 10) configuration described in the common test conditions [13] and decoded with the Fraunhofer HHI HEVC software decoder.

III. RESULTS ON A WORKSTATION CPU

All runtime measurements have been performed on the same type of computer which has an eight core Intel Xeon E5-2687W CPU running at 3.1GHz. Simultaneous Multithreading (SMT, also called Hyperthreading by Intel) is disabled to limit the number of hardware threads to eight and dynamic over-clocking (aka Turbo Boost) is disabled to have reproducible results.

Fig. 1 shows the speedup factor that can be achieved for different numbers of threads used for frame-level parallel decoding. It can be seen that the speedup for the Intra configurations increases compared to the Random Access speedup when the number of threads increases since all frames can be independently processed in parallel. Because the Random Access configuration uses inter-picture prediction, the frame-level parallelism provides a non-linear speedup. This is due to the fact that synchronization between the threads is more frequent to account for inter-picture prediction sample referencing. The speedup saturates when the number of worker threads reaches the maximum number of CPU cores which is eight. Only for the Random Access configurations, the speedup gets larger when the number of threads is further increased to ten. This can be explained by the initial, still

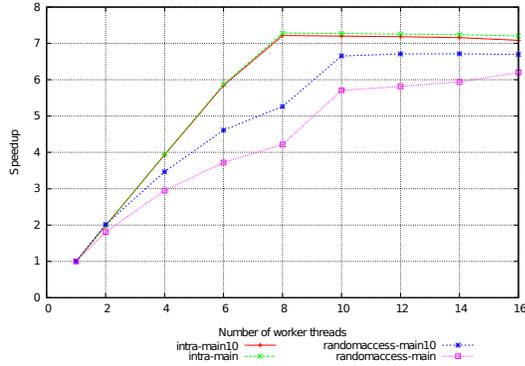


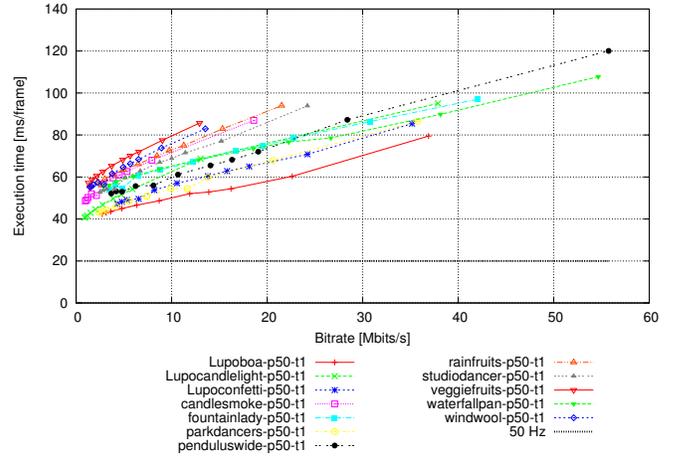
Fig. 1. Decoding speedup on an Intel Xeon E5-2687W workstation CPU at 3.1GHz averaged over the complete 4K EBU UHD-1 test set for Intra and Random Access configurations.

sub-optimal implementation of frame-level parallelism where the number of parallel processed frames is set equal to the number of worker threads. When a worker thread is idle, it cannot start decoding another picture when this would increase the number of simultaneously decoded frames. Especially for the hierarchical coding structure in the Random Access configuration, where frames inside the group of pictures (GOP) are coded with different quantization parameters, decoding times of frames vary much more than for the Intra configuration. Choosing more worker threads than CPU cores helps in these cases since it increases the number of frames that are allowed to be processed in parallel.

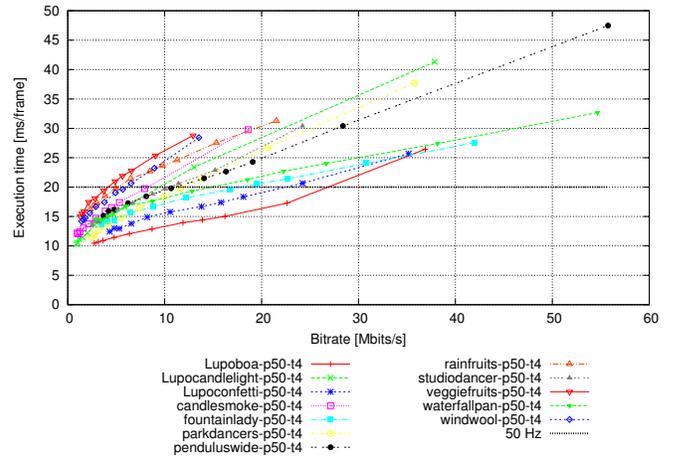
Going a bit more in the details for the Random Access Main 10 configuration with 10bit video, Fig. 2a, Fig. 2b and Fig. 2c show the execution time of the Fraunhofer HHI decoder for all the UHD-1 50Hz sequences when one, four and ten worker threads are used. According to Fig. 1, the performance peaks when using ten worker threads and saturates from this point on. The horizontal dashed line represents the real-time limit for 50Hz which is $\frac{1000}{50} = 20$ [ms/frame]. Whether real-time decoding is possible or not depends on the sequence and the bitrate. For example when four threads are used, Lupo boa can be decoded in real-time up to 27.5 Mbits/s while veggie fruits passes the 20 ms/frame line at 4 Mbits/s. Looking at the objective quality for the different sequences at different bitrates as shown in Fig. 3, it can be seen that Lupo boa provides a Peak Signal to Noise Ratio (PSNR) of 39.5 dB at 27.5 Mbits/s and veggie fruits already reaches 40 dB at 4 Mbits/s. Hence, real-time decoding for both sequences at a good objective quality is feasible using four threads on four cores.

IV. RESULTS ON A DESKTOP CPU

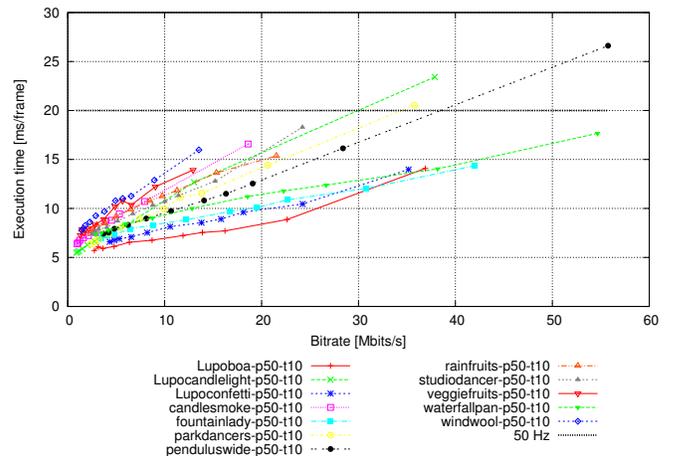
In addition to the Xeon workstation CPU, the Random Access Main 10 configuration bitstreams have also been decoded on a state-of-the-art four core Intel i7-2920XM desktop CPU running at 2.5GHz. This configuration is considered to be more representative for systems that people have at home. Here, SMT is enabled giving a maximum of eight



(a) Xeon E5-2687W workstation CPU at 3.1GHz using 1 core - 1 thread



(b) Xeon E5-2687W workstation CPU at 3.1GHz using 4 cores - 4 threads



(c) Xeon E5-2687W workstation CPU at 3.1GHz using 8 cores - 10 threads

Fig. 2. Decoding time for each sequence of the 4K EBU UHD-1 test set for the Random Access Main 10 configuration with 1, 4 and 8 cores.

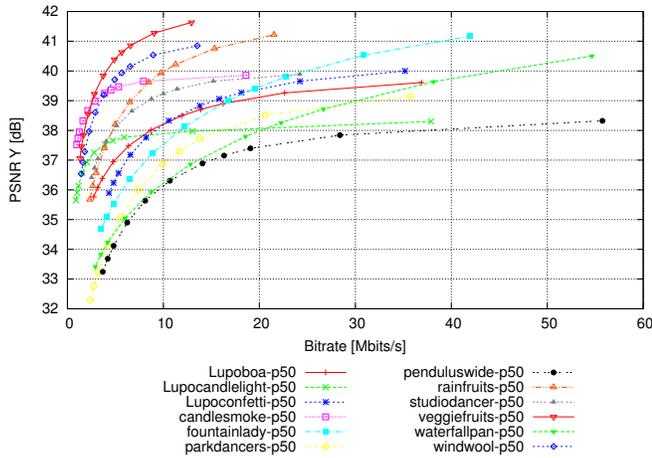


Fig. 3. Rate-distortion performance of the 4K EBU UHD-1 test set for the Random Access Main 10 configuration.

hardware threads for the software to use. As for the Xeon workstation CPU, Turbo Boost is disabled to not distort the runtime measurements by varying CPU clock rates.

Similarly to Fig. 1, Fig. 4 shows the speedup achieved when more than one worker thread is used. Although SMT provides eight hardware threads, the speedup when using more than four worker threads is not increased as much as it would be when having eight cores available. Therefore, the four additional hardware threads or virtual cores cannot be counted as full cores for frame-level parallel decoding.

Fig. 5a, Fig. 5b and Fig. 5c show the execution time over the bitrate for all EBU UHD-1 test sequences. It can be seen that the performance for one and four worker threads is comparable to the Xeon workstation CPU. In the best performing configuration, i.e. when all CPU resources are used with ten worker threads, all sequences can be decoded at least up to 10 Mbits/s. When mapping the maximum bitrates again to the PSNR values representing objective quality in Fig. 3, the coded bitstreams have at least a decent objective quality. The sequence pendulus wide for example, which has the worst coding performance according to Fig. 3, can be decoded in real-time up to 25 Mbits/s. At 25 Mbits/s, its PSNR value is around 37.5 dB which is quite good considering that the rate distortion curve saturates around 38 dB.

V. CONCLUSION

It has been shown that real-time software decoding of 4K 50Hz video with HEVC is feasible on current desktop CPUs using four CPU cores. Encoding 4K video in real-time on the other hand remains a challenge. Therefore, first use cases of 4K video coded with HEVC are expected to be limited to offline encoded material for internet services like video on demand.

REFERENCES

[1] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, 2003.

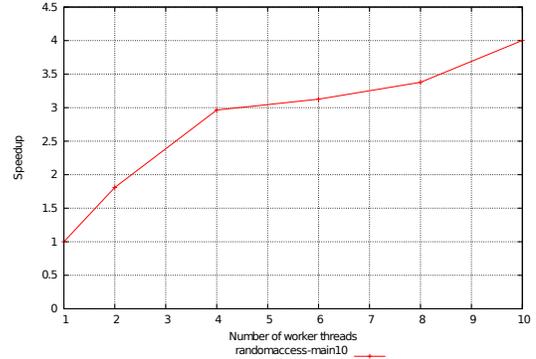


Fig. 4. Decoding speedup on an Intel i7-2920XM desktop CPU at 2.5GHz averaged over the complete 4K EBU UHD-1 test set for Random Access Main 10 configuration.

[2] B. Bross, W.-J. Han, J.-R. Ohm, G. J. Sullivan, Y.-K. Wang, and T. Wiegand, "High Efficiency Video Coding (HEVC) text specification draft 10 (for FDIS & Last Call)," document JCTVC-L1003 of JCT-VC, Geneva, CH, Jan. 2013.

[3] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

[4] J.-R. Ohm, G. J. Sullivan, H. Schwarz, T. K. Tan, and T. Wiegand, "Comparison of the Coding Efficiency of Video Coding Standards Including High Efficiency Video Coding (HEVC)," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1669–1684, Dec. 2012.

[5] Y. J. Ahn, W. J. Han, and D. G. Sim, "Study of decoder complexity for hevc and avc standards based on tool-by-tool comparison," in *Proceeding of SPIE 8499, Applications of Digital Image Processing XXXV*, October 2012, p. paper 84990X.

[6] F. Bossen, B. Bross, K. Sühring, and D. Flynn, "HEVC Complexity and Implementation Analysis," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, pp. 1669–1684, Dec. 2012.

[7] M. Alvarez-Mesa, C. C. Chi, B. Juurlink, V. George, and T. Schierl, "Parallel Video Decoding in the Emerging HEVC Standard," in *Proceedings of the 37th International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, March 2012.

[8] C. C. Chi, M. Alvarez-Mesa, B. Juurlink, V. George, and T. Schierl, "Improving the Parallelization Efficiency of HEVC Decoding," in *Proceedings of IEEE International Conference on Image Processing (ICIP)*, Oct 2012.

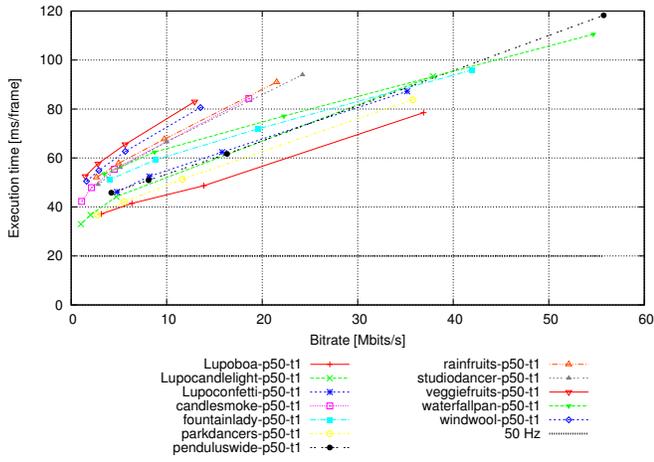
[9] C. C. Chi, M. Alvarez-Mesa, B. Juurlink, G. Clare, F. Henry, S. Pateux, and T. Schierl, "Parallel Scalability and Efficiency of HEVC Parallelization Approaches," *IEEE Transaction of Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1827–1838, Dec. 2012.

[10] C. C. Chi, M. Alvarez-Mesa, J. Lucas, B. Juurlink, and T. Schierl, "Parallel HEVC Decoding on Multi- and Many-core Architectures," *Journal of Signal Processing Systems*, pp. 1–14, Dec. 2012.

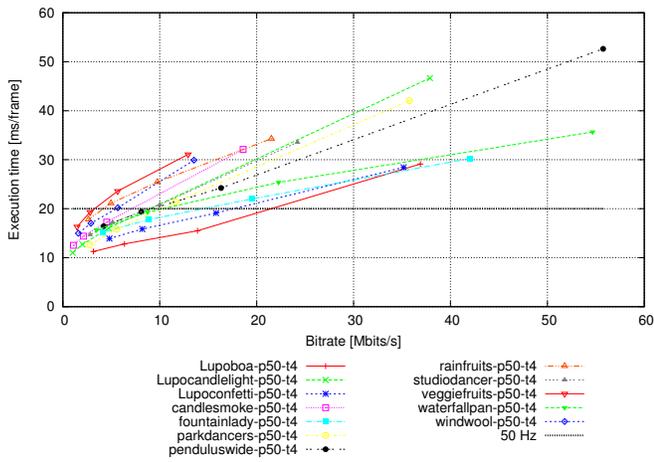
[11] European Broadcast Union, "EBU UHD-1 Test Set," 2013. [Online]. Available: <http://tech.ebu.ch/testsequences/uhd-1>

[12] JCT-VC, "Subversion Repository for the HEVC Test Model version HM10," 2013. [Online]. Available: https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/tags/HM-10/

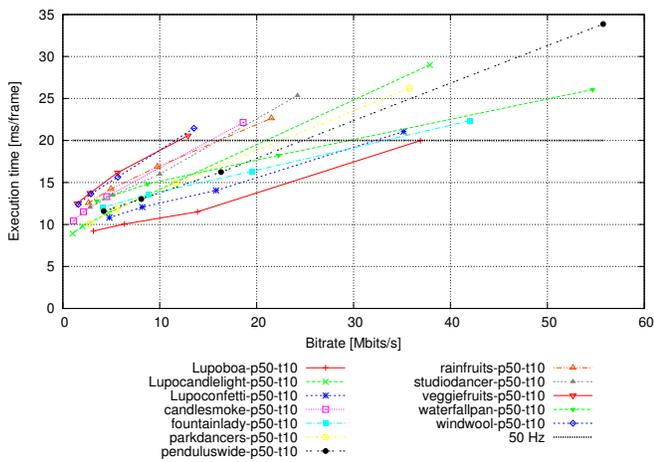
[13] F. Bossen, "Common HM test conditions and software reference configurations," document JCTVC-L1100 of JCT-VC, Geneva, CH, Jan. 2013.



(a) Intel i7-2920XM desktop CPU at 2.5GHz using 1 core - 1 thread



(b) Intel i7-2920XM desktop CPU at 2.5GHz using 4 cores - 4 threads



(c) Intel i7-2920XM desktop CPU at 2.5GHz using 4 cores - 10 threads

Fig. 5. Decoding time for each sequence of the 4K EBU UHD-1 test set for the Random Access Main 10 configuration with 1 and 4 cores.