A Benchmark of Entropy Coders for the Compression of Genome Sequencing data

S. Casale-Brunet
P. Ribeca
C. Alberti
U. Ozturk
M. Mattavelli

DCC 2022

Compression of Short NGS Reads

- Short next-generation sequencing (NGS) reads are highly typically sequenced with high redundancy
  - x20-50 coverage of the original genome is typical
- Sequencing reads contain three fields to be compressed:
  - Read names
  - Quality values
  - Nucleotide sequences
- Nucleotide sequences are easy to compress due to the redundancy arising from high coverage sequencing
- Quality values and read names are difficult to compress due to their noisy nature
  - These typically form >50% of the content of compressed files

MPEG-G

- ISO/IEC 23092
- Open standard for the coding genomic data
- Provides a syntax framework for interoperable decoding processes

- File is organized into dataset group & datasets

- Unit of granular access are called access units containing descriptors

- Genomic sequencing data is organized into descriptors:
  - Descriptors contain all the necessary information for the reconstruction of genomic reads or alignments

MPEG-G

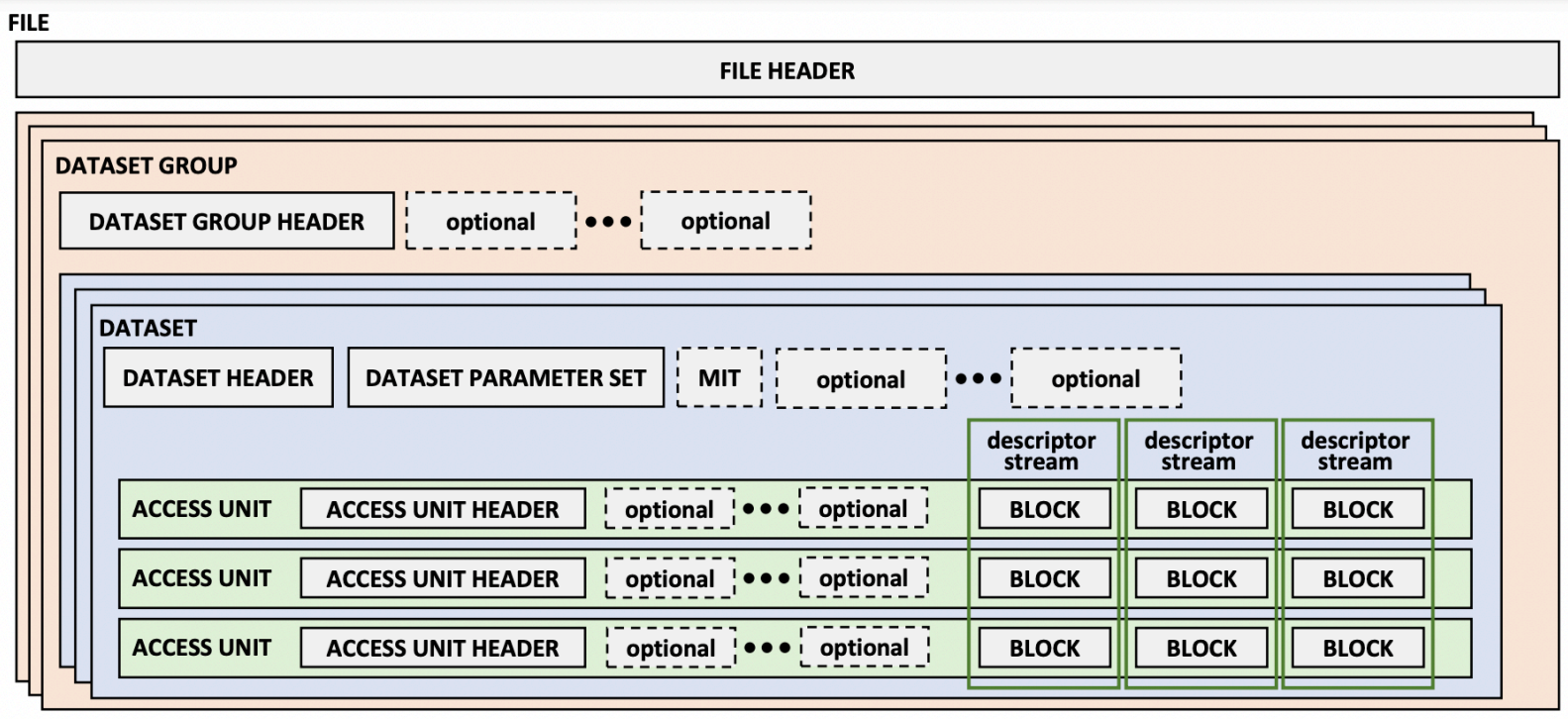


Figure 1: MPEG-G File organisation

MPEG-G Compression of Reads

- Second Edition of MPEG-G only supports **C**ontext **A**daptive **B**inary **A**rithmetic **C**oding (CABAC) for the compression of genomic descriptors
- CABAC is utilised in other MPEG standards (e.g. ISO/IEC 14496) and is very efficient in the video coding
  - Provides high compression rates
- In the context of coding genomic data, CABAC has some limitations:
  - Read names and quality values are noisy, and hence are not as easily compressible
  - CABAC is difficult to parallelise or vectorise

MPEG-G Compression of Reads

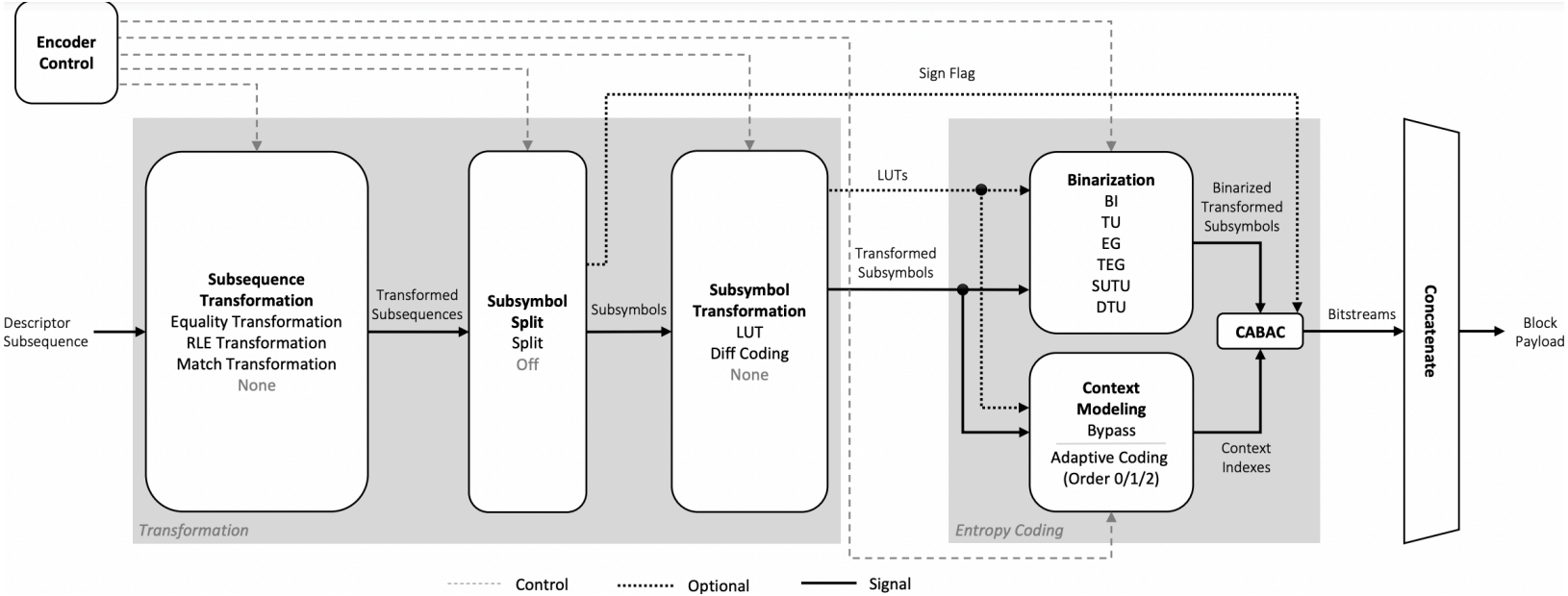


Figure 2: CABAC Transformation and Entropy Coding

MPEG-G Compression of Reads

- CABAC takes longer times compared to other entropy coders to encode descriptors coding read names and quality values
  - For comparatively little gain in compression rates due to the noisy nature of read names and quality values

- The MPEG-G standard could benefit from supporting low complexity entropy coders
  - Better tradeoff between compression rates and times
  - Sacrifice some compression rate for significant gain in time

Low Complexity Coder Benchmarks

- Our Contribution: A benchmark of compression-decompression speeds and compression rates of a set of entropy coders
- Two types of input data:
  - Raw FASTQ: Data gathered in string form directly from FASTQ files
  - MPEG-G uncompressed bitstreams: Data gathered by decompressing a compressed MPEG-G streams:
    - These samples contain both aligned and unaligned data
- Input files are derived from ERR174310.chr9 and G15111.HCC1143.BL.1.chr9 on the MPEG-G database

Low Complexity Coder Benchmarks: Methodology

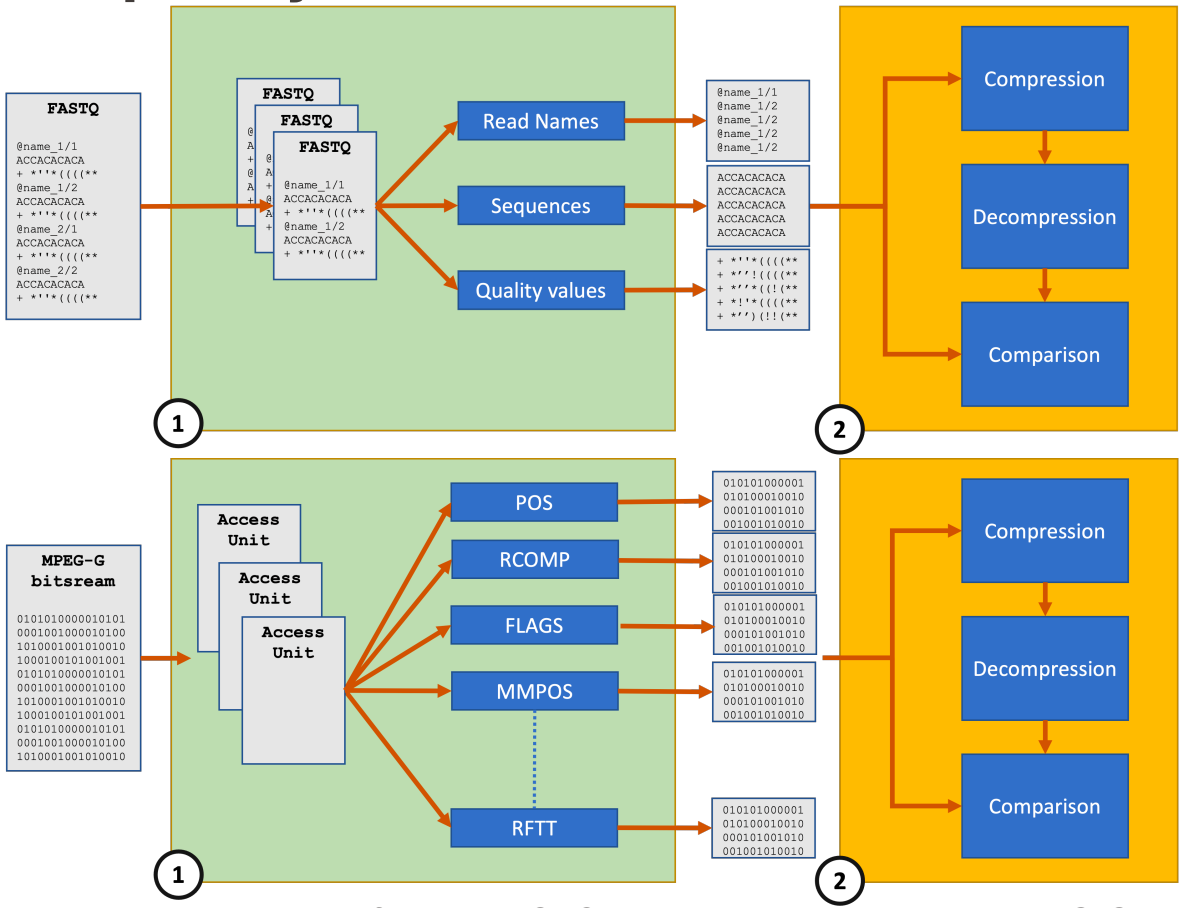
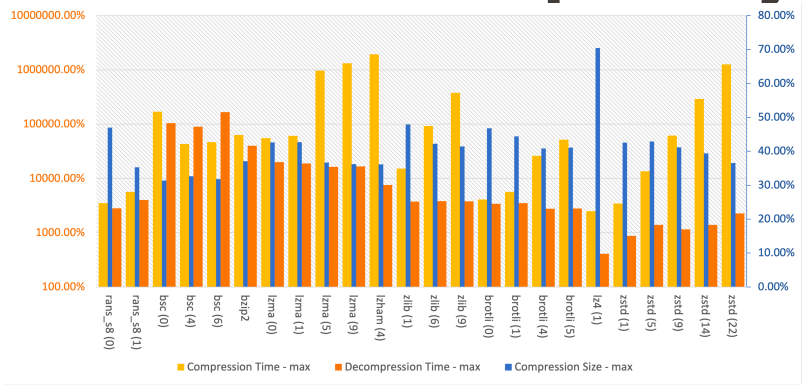


Figure 3: Analysis methodology for raw FASTQ data and uncompressed MPEG-G bitstreams

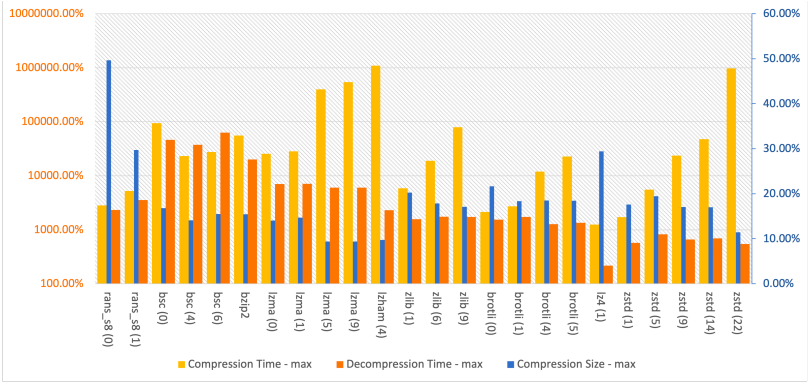
Low Complexity Coder Benchmarks

- Benchmarked entropy coders:
  - rANS (range asymmetric numeral system)
  - bsc (block-sorting compression)
  - bzip2
  - LZMA
  - LZHAM
  - zlib
  - LZ4
  - brotli
  - Zstandard

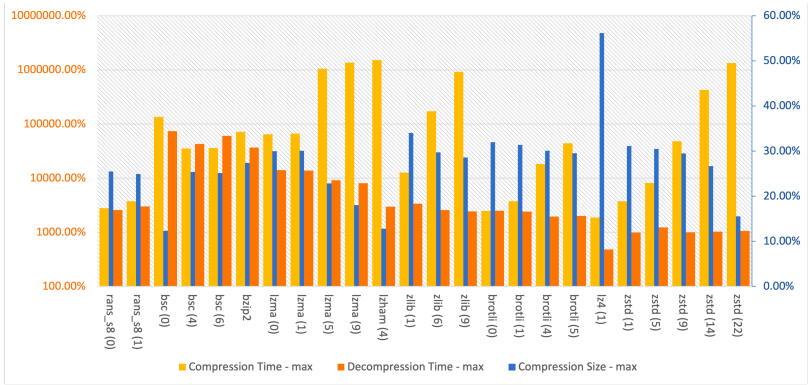
Low Complexity Coder Benchmarks



Quality values, ERR174310_chr9_1 entire file

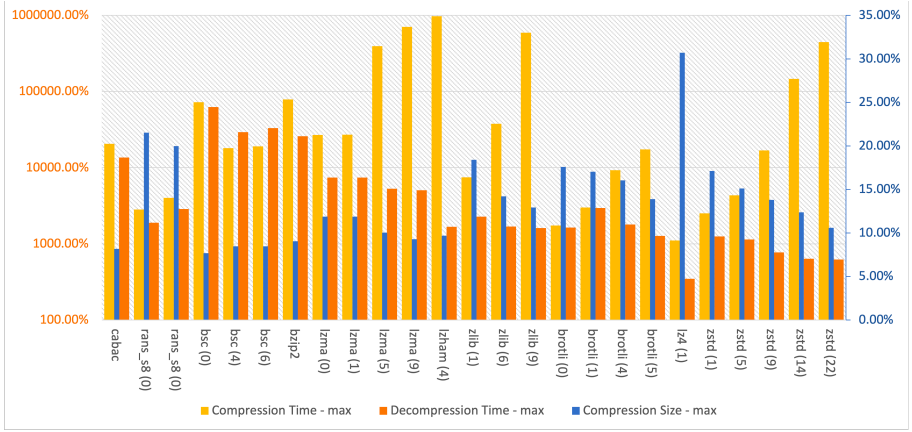


Read names, ERR174310_chr9_1 entire file



Nucleotide sequences, ERR174310_chr9_1 entire file

Low Complexity Coder Benchmarks



Uncompressed MPEG-G bitstream, ERR174310_chr9_1 entire file

Descriptor	Size	
	Bytes	Percentage
POS	6785529	01.0
RCOMP	76954	00.0
FLAGS	103692	00.0
MMPOS	4369137	00.7
MMTYPE	989515	00.2
CLIPS	257455	00.0
UREADS	0	00.0
RLEN	6316182	01.0
PAIR	9908072	01.5
MSCORE	2556072	00.4
MMAP	0	00.0
MSAR	0	00.0
RTYPE	0	00.0
RGROUP	0	00.0
QV	542275751	82.2
RNAME	85735014	13.0
RFTP	0	00.0
RFTT	0	00.0

(a) Item 43

Descriptor	Size	
	Bytes	Percentage
POS	0	00.0
RCOMP	0	00.0
FLAGS	0	00.0
MMPOS	0	00.0
MMTYPE	0	00.0
CLIPS	0	00.0
UREADS	703106	46.8
RLEN	10607	00.7
PAIR	101	00.0
MSCORE	0	00.0
MMAP	0	00.0
MSAR	0	00.0
RTYPE	0	00.0
RGROUP	0	00.0
QV	683051	45.5
RNAME	104290	06.9
RFTP	0	00.0
RFTT	0	00.0

(b) Item 47

Descriptor contribution to the overall bitstream size

Low Complexity Coder Benchmarks: Results

- Complete and detailed results, scripts, and plots are available on:
  - https://github.com/epfl-scistimm/2022-DCC
- Key takeaways from results:
  - CABAC provides higher compression rates, but slow decompression speeds
  - LZ4 provides low compression rates, but is extremely fast both in compression and decompression
  - Zstandard and brotli were the closest codecs to the Pareto frontier in terms of compression-decompression speeds and compression rates
  - bsc provides high compression rates

Low Complexity Coder Benchmarks: Results

- Choose entropy coder according to the use case:
  - High throughput: LZ4
  - Archival purposes: bsc
  - General/generic compression: zSTD
  - Different kinds of genomic data can be compressed with different coders
- Benchmark results were used to propose the integration of support for various entropy coders to the third edition of the MPEG-G standard