FastLanes on GPU: Analysing Data-Parallelized Compression Schemes

Azim Afroozeh [1], Lotte Felius [1], and Peter Boncz [1]

[1] Database Architectures, Centrum Wiskunde & Informatica, Netherlands
{azim,felius,boncz}@cwi.nl

High-performant analytical workloads can be accelerated by exploiting the massive parallelism provided by GPUs [Fang et al.(2010), Shanbhag et al.(2022)]. However, the main bottleneck of GPUs is their limited memory capacity. This limitation can be mitigated by using compression to fit more data into GPU memory and speed up data transfer between the CPU and GPU. Nonetheless, typical compression schemes used in database systems such as FOR, RLE and Delta encoding are challenging to parallelize on GPUs. Without parallelization however, all threads are blocked on the GPU until data is entirely decompressed.

Recent work by Shanbhag *et al.* proposes `GPU-FOR`, `GPU-DFOR` and `GPU-RFOR` to parallelize typical compression schemes. In their work, they assign each compressed value to a single thread. This adaptation introduces significant computational overhead during decompression. Alternatively, FastLanes [Afroozeh and Boncz(2023)] proposes data-parallelized layouts such as a *value interleaving technique* for bit-(un)packing and a *transposed layout* for DELTA and RLE encodings. These layouts enable CPUs to decompress data in parallel without extra computational overhead.

We argue that we can further improve the performance of decompression on GPUs by using the data parallelized layouts proposed by FastLanes. In our current work, we implemented FastLanes schemes on the GPU. Further, we compared the current state-of-the-art encoding schemes (1) `GPU-FOR`, (2) `GPU-DFOR` and (3) `GPU-RFOR` by Shanbhag *et al.* to the performance of decoding FOR, RLE and Delta on the GPU using the FastLanes layout. Our preliminary results show an improvement of $\sim$**20%** for bit-unpacking with FastLanes, and a $\sim$**2x** speedup for both Delta and RLE encodings. For future research, our aim is to optimize the FastLanes layout to adapt to both CPUs and GPUs and try to integrate these optimizations into a new file format.

References

[Afroozeh and Boncz(2023)] Azim Afroozeh and Peter Boncz. 2023. The FastLanes Compression Layout: Decoding 100 Billion Integers per Second with Scalar Code. *Proceedings of the VLDB Endowment* 16, 9 (2023), 2132–2144.

[Fang et al.(2010)] Wenbin Fang, Bingsheng He, and Qiong Luo. 2010. Database compression on graphics processors. *Proceedings of the VLDB Endowment* 3, 1-2 (2010), 670–680.

[Shanbhag et al.(2022)] Anil Shanbhag, Bobbi W Yogatama, Xiangyao Yu, and Samuel Madden. 2022. Tile-based lightweight integer compression in GPU. In *Proceedings of the 2022 International Conference on Management of Data*. 1390–1403.