Genome Read In-Memory (GRIM) Filter: Fast Location Filtering in DNA Read Mapping using Emerging Memory Technologies

Jeremie Kim[1], Damla Senol[1], Hongyi Xin[2], Donghyuk Lee[1,3], Mohammed Alser[4], Hasan Hassan[5], Oguz Ergin[5], Can Alkan[4] and Onur Mutlu[6,1]

[1] Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA, USA
[2] Department of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA
[3] NVIDIA Research, Austin, TX, USA
[4] Department of Computer Engineering, Bilkent University, Bilkent, Ankara, Turkey
[5] Department of Computer Engineering, TOBB University of Economics and Technology, Söğütözü, Ankara, Turkey
[6] Department of Computer Science, Systems Group, ETH Zürich, Zürich, Switzerland

In the past couple of decades, high-throughput sequencing (HTS) technology has resulted in a massive influx of available genetic data. Using HTS technology, genomes are sequenced relatively quickly and result in many short DNA read sequences (reads) that are used to analyze the donor's genome. Due to the enormous size and complexity of the human genome, the analysis of the resulting short reads still takes days to complete even if we use state-of-the-art methods. The first step of genome analysis, *read mapping*, determines the origins of billions of reads within a reference genome to identify the donor's genomic variants. Hash-table based read mappers are a common type of comprehensive read mappers. They operate by fetching from a pre-generated hash-table potential mapping locations of a read in the reference genome, which are verified by local alignment, a computationally-expensive dynamic programming algorithm that determines similarity between the read and the potential mapping segment of the reference genome. Alignment has traditionally been the computational bottleneck of read mapping [1,2], but recently, many works have been proposing a new step called Location Filtering in order to alleviate this bottleneck.

Location Filtering is a critical step where many incorrect potential locations from the hash-table are discarded before local alignment verifies such locations. FastHASH [1], SHD [2], and GateKeeper [3] propose variations of Location Filtering that achieve the same goal of discarding only incorrect locations to reduce end-to-end runtime of hash-table based read mapping. Location Filtering is now the computational bottleneck of read mapping [1,2,3].

Our goal is to create an efficient Location Filter that quickly discards as many false negative locations as possible before alignment, while retaining a zero false positive rate (i.e., removing candidate locations in the reference genome with low similarity to a read while keeping all locations with high similarity). Efficiently filtering incorrect mappings before alignment significantly improves throughput and latency of hash-table based read mapping. We propose a novel filtering algorithm that quickly eliminates from consideration reference genome segments where alignment would yield no matches. Our algorithm's novelty mainly stems from its design to exploit 3D-stacked memory systems. 3D-stacked memory is an emerging technology that tightly integrates computation and high-capacity memory in a single die stack, thereby enabling concurrent processing of large data chunks at low latency and high bandwidth. The key ideas of our design consist of 1) a new representation of coarse-grained segments of the reference genome such that the genome can be operated on in parallel in large chunks using simple bitwise operations and 2) exploiting the parallel computation capability of 3D-stacked memory to run massively-parallel in-memory operations on the new genome representation. We call our resulting filter the GRIM-Filter.

In this work, we show how GRIM-Filter can be used with any hash-table based read mapping algorithm and how it effectively exploits processing-in-memory capabilities of 3D-stacked memory. We show that when running with 5% error tolerance, GRIM-Filter results in *5.59x-6.41x* less false positive locations and provides a *1.81x-3.65x* end-to-end speedup over the state-of-the-art read mapper mrFAST with FastHASH [1].

[1] Xin, Hongyi, et al. "Accelerating read mapping with FastHASH." *BMC Genomics*, 14(Suppl 1):S13, (2013).
[2] Xin, Hongyi, et al. "Shifted Hamming distance: a fast and accurate SIMD-friendly filter to accelerate alignment verification in read mapping." *Bioinformatics*, page 856, (2015).
[3] Alser, Mohammed, et al. "GateKeeper: Enabling Fast Pre-Alignment in DNA Short Read Mapping with a New Streaming Accelerator Architecture." *arXiv preprint arXiv:1604.01789* (2016).