Accelerating Approximate Pattern Matching with Processing-In-Memory (PIM) and Single-Instruction Multiple-Data (SIMD) Programming

Damla Senol Cali[1], Zülal Bingöl[2], Jeremie Kim[1,3], Rachata Ausavarungnirun[1], Saugata Ghose[1], Can Alkan[2] and Onur Mutlu[3,1]

[1] Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA, USA
[2] Department of Computer Engineering, Bilkent University, Bilkent, Ankara, Turkey
[3] Department of Computer Science, ETH Zürich, Zürich, Switzerland

High throughput sequencing (HTS) technology enables fast and inexpensive generation of billions of short DNA sequences (i.e., reads) from a genome [1,2]. Unfortunately, performing HTS data analysis introduces significant computational challenges. Therefore, we need computational techniques that can quickly and accurately process and analyze this data. The first step in analyzing HTS data is *mapping*, a process that determines the original location of each read in the genome. Briefly, read mapping is finding each read's best match within a long *reference genome.* The last step of most read mapping algorithm is *verification,* which typically consists of expensive dynamic programming algorithms (e.g., Needleman-Wunsch algorithm [3]), where over 90% of the execution time is spent. Many prior studies [4, 5, 6, 7] have identified this bottleneck in mapping and have proposed numerous methods for accelerating this expensive step.

Verification is essentially an application of approximate string matching problem, and thus can benefit from existing techniques used to optimize general-purpose string matching. Our goal in this work is to replace the computationally-expensive dynamic programming algorithm used for verification with the bitap algorithm used for fuzzy search [8]. Bitap is well-suited for verification because it can perform approximate string matching with fast and simple bitwise operations, but it introduces a number of new computational challenges on existing systems. For example, the operations used during bitap can be performed in parallel, but high-throughput parallel bitap computation requires a large amount of memory bandwidth that is currently unavailable to the processor. To tackle these challenges, we propose two ideas: (1) using single-instruction multiple-data (SIMD) programming to take advantage of the high amount of parallelism available in the bitap algorithm, and (2) performing processing-in-memory (PIM) to exploit the high internal bandwidth available inside new and emerging memory technologies.

We observe that bitap is both PIM- and SIMD-friendly since it is entirely based on bitvectors and elementary bitwise operations. Here we report our recent work on extending *bitap* for PIM and SIMD programming in order to accelerate the algorithm and therefore accelerate the full verification step by replacing the slow and expensive dynamic programming algorithm with accelerated bitap algorithm.

REFERENCES

[1] Alkan, Can, Saba Sajjadian, and Evan E. Eichler. "Limitations of next-generation genome sequence assembly." Nature Methods 8.1 (2011): 61.
[2] Van Dijk, Erwin L., et al. "Ten years of next-generation sequencing technology." Trends in Genetics 30.9 (2014): 418-426.
[3] Needleman, Saul B., and Christian D. Wunsch. "A general method applicable to the search for similarities in the amino acid sequence of two proteins." Journal of Molecular Biology 48.3 (1970): 443-453.
[4] Alser, Mohammed, et al. "GateKeeper: a new hardware architecture for accelerating pre-alignment in DNA short read mapping." Bioinformatics 33.21 (2017): 3355-3363.
[5] Kim, Jeremie S., et al. "GRIM-Filter: fast seed location filtering in DNA read mapping using Processing-in-Memory technologies." arXiv preprint arXiv:1711.01177 (2017).
[6] Wilton, Richard, et al. "Arioc: high-throughput read alignment with GPU-accelerated exploration of the seed-and-extend search space." PeerJ 3 (2015): e808.
[7] Myers, Gene. "A fast bit-vector algorithm for approximate string matching based on dynamic programming." Journal of the ACM (JACM) 46.3 (1999): 395-415.
[8] Baeza-Yates, Ricardo, and Gaston H. Gonnet. "A new approach to text searching." Communications of the ACM 35.10 (1992): 74-82.