Boosting Backward Search Throughput for FM-Index Using a Compressed Encoding

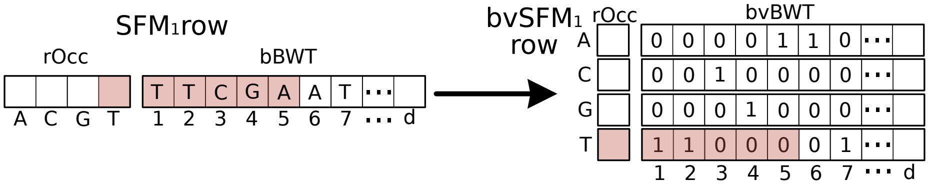
Jose M. Herruzo*, Sonia González-Navarro*, Pablo Ibáñez†, Victor Viñals†,
Jesús Alastruey-Benedé†, and Oscar Plata*

*Dept. Computer Architecture
University of Malaga, Spain
{jmherruzo,sgn,oplata}@uma.es

†Dept. Computer Science & Systems Engineering
University of Zaragoza, Spain
{imarin,victor,jalastru}@unizar.es

The rapid development of DNA sequencing technologies has demanded for compressed data structures supporting fast pattern matching queries. FM-index is a widely-used compressed data structure that also supports fast pattern matching queries. It is common for the exact matching algorithm to be memory bound, resulting in poor performance. Searching several symbols in a single step improves data locality, although the memory bandwidth requirements remains the same.

We propose a new data-layout of FM-index, called Split bit-vector, that compacts all data needed to search k symbols in a single step (k-step), reducing both memory movement and computing requirements at the cost of increasing memory footprint. The original sampled FM-Index uses a data structure containing a set of counters for each bucket together with the reference text. Our new data layout divides each row of SFM into a new row per character in the alphabet and the compression of each bucket (of size d) using a bitmap where each symbol is represented by a single bit.



Experiments have been carried out on an Intel Xeon Phi KNL processor, using all the available cores, the vector AVX-512 support and the 3D MCDRAM memory, resulting in a solution 3x faster than previous GPU implementations, and 25x faster than the fast *sdsl-lite* library in KNL.

Implementation	Throughput (LFMs/s)	Index size (GB)
2-Step Split bit-vector (d=64)	12.0 G	12
2-Step FM-index (d=16)	5.1 G	13.5
sdsl-lite library on KNL	0.455 G	1.25
2-Step FM-index on Kepler GTX Titan GPU*	3.8 G	3
NVIDIA NVBIO on Tesla P100	2.7 G	0.23

* doi:10.1109/TCBB.2014.2377716

LFM: Last-to-First Mapping