* Tool name

Ramp, Maq, Shrimp, Bowtie, BWA, Soap2

* Technology used

Burrow Wheel and Cushaw

* Approach
* 1. Burrow Wheel Transform
* 2.Searching for exact matches
* 3.Searching for inexact matches
* 4. Locating occurrences
* 5.Mapping in Cuda
* 6.Pair-end Mapping
* Implementation detail

1. Burrow Wheel Transform

A BWT is a reversible permutation of

the text. For a genome sequence G defined over - = {A,C,G,T}, the forward

BWT of G can be constructed in three steps. First, a special character $, which

is lexicographically smaller than any character in -

, is appended to the end of

G to form a new sequence G$. Second, a conceptual matrix M is constructed,

whose rows are all cyclic rotations of G$ (equivalent to all suffixes of G)

sorted in lexicographical order and each column is a permutation of G$.

Finally, the transformed text L (i.e. the forward BWT of G) is formed by

taking the last column of M . A suffix array SA, where SA[i] stores the

the starting position of the ith smallest suffix of G can be constructed from M

using the one-to-one correspondence relationship between SA[i] and the ith

row of M.

2. Searching for exact matches-

A sequence S that is a substring of G, each occurrence of S can be found using a

backward search procedure based on the FM index.

1. Searching for inexact matches

The search for inexact matches can be transformed into the search for exact matches of

all permutations of all possible bases at all positions of a short read.All inexact matches

can be found by traversing all paths using either depth-first search (DFS) or breadth-first search (BFS) approaches.

|  |  |
| --- | --- |
|  | 3. Locating occurrences-  After getting the SA interval, the position of each occurrence in G can be  determined by directly looking up the SA   * Performance Analysis-   For the paired-end alignment, CUSHAW achieves significant  speedups over all other three aligners (with the exception that for  SRR034966, CUSHAW on two GPUs executes only 1.3× faster  than Bowtie on four CPU cores). On a single GPU (two GPUs),  CUSHAW achieves an average speedup of 5.7 (2.4) with a highest of  11.8 (4.7) over Bowtie, an average speedup of 8.3 (5.5) with a highest  of 14.5 (12.2) over BWA and an average speedup of 8.5 (4.1) with a  highest of 24.3 (10.4) over SOAP2, where the three aligners run on  a single CPU core (four CPU cores) for all five datasets. Similar to  the single-end alignment.   * Advantage-  1. Alignment quality   The alignment quality is conventionally evaluated by computing  how many single-end reads are found to match to the  reference genome and how many paired-end reads are paired  together using simulated or real short-read datasets. For the single-end alignment,  CUSHAW is inferior to both Bowtie and SOAP2 for all datasets.  CUSHAW aligned more reads than BWA for the SRR002273 and  ERR000589 datasets, but aligned fewer for the other three datasets.  2. SNP Calling  3. Execution speed  Besides alignment quality, another major concern about short read  alignment is the execution speed considering the sheer volume  of short reads produced from the high-throughput sequencing  technologies. For the paired-end alignment, CUSHAW achieves significant  speedups over all other three aligners (with the exception that for  SRR034966, CUSHAW on two GPUs executes only 1.3× faster  than Bowtie on four CPU cores). On a single GPU (two GPUs),  CUSHAW achieves an average speedup of 5.7 (2.4) with a highest of  11.8 (4.7) over Bowtie, an average speedup of 8.3 (5.5) with a highest  of 14.5 (12.2) over BWA, and an average speedup of 8.5 (4.1) with a  highest of 24.3 (10.4) over SOAP2, where the three aligners run on  a single CPU core (four CPU cores) for all five datasets. Similar to the single-end alignment.   * Disadvantage-   The performance of SNP calling from short  read alignments were also examined. Although the single example  presented is insufficient to fully evaluate the performance of all  the aligners, it still sheds some light on the impact of the different  aligners in terms of their SNP calling performance.  At present, CUSHAW only supports ungapped alignment for  single-end and paired-end reads, where it supports a maximal read  length of 128 by default (can be configured up to 256) and a  maximal genome length of 4 billion bases. For longer reads that  tend to contain indels, the introduction of gapped alignment might  be able to increase the probabilities that reads are matched to the  reference genome. CUSHAW outputs the aligned (or paired) reads  in the SAM format (Li H. et al., 2009) to take advantage of the  SAMtools software package to facilitate the downstream analysis  of alignments. The major challenges for short read alignment using  CUDA are the frequent accesses to global memory with poor data  locality and the divergence of alignment paths for different short  reads. The poor data locality will lead to more misses in the L1/L2  caches for global memory accesses, and the divergence of alignment  paths causes the execution paths of the threads in a warp to diverge  frequently. |