Minimal perfect hash functions in large scale bioinformatics Problem
Antoine Limasset, Camille Marchet, Pierre Peterlongo, Lucie Bittner

To cite this version:
Antoine Limasset, Camille Marchet, Pierre Peterlongo, Lucie Bittner. Minimal perfect hash functions in large scale bioinformatics Problem. JOBIM 2016, Jun 2016, Lyon, France. <hal-01341718>
Minimal perfect hash functions in large scale bioinformatics

Antoine Limasset, Camille Marchet and Pierre Peterlongo
INRIA/IRISA/GenScale, Campus de Beaulieu, 35042 Rennes Cedex, France
antoine.limasset@irisa.fr, camille.marchet@irisa.fr, pierre.peterlongo@irisa.fr

(Meta)Genomic Data
Billions of short sequences of hundreds of base pairs, from one or multiple genomes

Questions
Dataset comparison: Detection of similar reads inter or intra datasets

Problem
Indexing huge set of elements

Hash functions
- Classical hashing
- Perfect hashing
- Minimal perfect hashing

BBhash library
- Memory efficient (less than 3 bits per key)
- Fast query (200ns)
- Fast to construct (even for billions elements)

Quasi-dictionary
- Put a fingerprint in the value and check it at the query
- False positive rate:
  $\frac{2^{k+1} - f}{2^{k+1}} - \frac{2}{2^f}$
- Memory consumption:
  Bit/elements: 10 FP rate: $1/10^2$
  Bit/elements: 20 FP rate: $1/10^5$
  Bit/elements: 30 FP rate: $1/10^8$

Short Read Connector tools
- SRC Linker: Output reads in A that has T Kmers that appear in set B.
  Kmer $\rightarrow$ MPHe
- SRC Linker Check: Output reads in A that has T Kmers in common with the reads of B and estimate their coverage.
  Kmer $\rightarrow$ MPHe

Results
- Time for indexing and querying 1M reads with SRC Linker
- Time for indexing and querying 100M reads with SRC Count
- Memory used for indexing 100M reads with SRC Count

Less pressure on your machine!

References
- Bowtie2
- BWA
- Starcode
- BLAST
- SRC

Links
- Bbhash library: github.com/rizkg/BBHash
- Quasi-dictionary: github.com/pierrepeterlongo/quasi_dictionary
- Short Read Connector: github.com/GATB/rconnector