1. Alignment of Whole Genomes
2.Fast Algorithms for Large-scale Genome Alignment and Comparison

Chen, Pai-Hsuen

May 5, 2004

outline

Suffix Tree

Papers

MUMmer web site
http://www.tigr.org/software/mummer/

Suffix Tree Introduction :
http://www.csse.monash.edu.au/~ lloyd/tildeAlgDS/Tree/Suffix

Building a suffix tree:
http://pauillac.inria.fr/~ quercia/documents-info/Luminy-98/albert/JAVA+html/SuffixTreeGrow.html

Building a suffix tree in linear time and space:
http://www.iis.sinica.edu.tw/~ hil/bioinfo/2001/lect3.ppt

Applications:
http://www.iis.sinica.edu.tw/~ hil/bioinfo/2001/lect4.ppt

Applications:

Find all occurrences of P as a substring of T

Longest substring common to a set T of strings

Find the longest palindrome in S

Alignment of whole genomes
Nucleic Acids Research, 1999

http://www.tigr.org/software/mummer/MUMmer.pdf

Problem:
Sequences are closely related.
Millions of nucleotides in length

Goal:
All single necleotide polymorphisms(SNPs)
large inserts,
significant repeats,
tandem repeats,
reversals, and
exact matching.

Algorithmic Methods:

- Suffix tree
- Longest Increasing subsequence(LIS)
- Smith-Waterman alignment

Details in alignment

Perform a Maximal unique match(MUM) (Fig. 1)
suffix tree: McCreight's Algorithm
Linear in time and space($<$ 37 bytes/base)
Constructing a suffix tree T for genome A and then adding the suffixs for genome B to T
All MUMs can be identified in a single scan.
The length of the shortest MUM that the system will identify.

Sort the matches and find the longest possible set of matches(LIS)
$O(klogk)$ (Fig 3)

Close the gaps

- SNP: surrounded by MUM subsequences
- Insert detection: larget gaps in the alignment one genome and not the other
- Polymorphic regions: Smith-Waterman for short length Recursively with reduced mininum MUM length
- Repeat processing: (Fig 5):
  Two interval overlaps:(0..13, 10..25 in Genome A)
  difference in overlap in the two genomes(8 in B and 4 in A)

experiments :

Comparing two strains of tuberculosis:
H37Rv
CDC1551

http://www.tigr.org/tigr-scripts/CMR2/webmum/mumplot

on DEC alpha 4100
5 sec for suffix tree
45 sec for LIS
5 sec for Smith-Waterman alignment

Comparing M.genitalium(580074 nt) and M. pneumoniae(816394 nt)

Fig. 7

on DEC alpha 4100
6.5 sec for suffix tree
0.02 sec for LIS
116 sec for Smith-Waterman alignmen

FASTA: many hours

Comparing human chromosome 12p13 and mouse chromosome 6

Fig 8

29 sec of CPU time
minimal length: 15

Fast algorithms for large-scale genome alignment and comparison

http://www.tigr.org/software/mummer/MUMmer2.pdf

Nucleic Acids Research 2002

MUMmer 2:

- speed and memory usage : a factor of nearly three

- make suffix tree for the reference and stream the query FIG.1

- matches clustering: pairs of matches that are sufficiently close and on sufficiently similar diagonals in the alignment matrix

NUCmer(nucleotide MUMmer)

create a map of all contig position within each of the multi-fasta files

concatenates the two files between the two genomes

runs MUMmer to find all exact matches between the two genomes

Table 1.

a partial sequenced genome with respect to a reference species

PROmer

translate DNA to amino acids in all six reading frames
compare each sequence in the first file to all in the second file

the frame with the best matches between the two input genomes
is preferred.