Approximate Overlap Finder (AOF)

Introduction

This is an implementation for a solution to exact and approximate all-pairs suffix prefix using compact prefix tree and Pigeon Hole principle respectively. We called our solution approximate overlap finder (AOF). This library improves and combines our previously presented techniques in order to offer a flexible tool for a de novo assembler engineer in order to handle the all-pairs suffix prefix problem efficiently. This solution uses OpenMp to support multithreading. This library is tested on Linux and Cygwin.

To compile: make (run make clean if necessary)

To simplify the usage of our functions, we provide one example (AOF.cpp) to demonstrate the usage of our functions. Feel free to modify it according to your needs. Assembler's engineers can simply choose to find exact or approximate overlaps, all of them or just the maximum, and with the preferred output format.

Functions

Our implementation depends on two main functions:

```
void APSP_PrefixTree(char *filename, char *output, int threads, int min);
```

This function finds the maximum exact overlap or all overlaps between every ordered pair of sequences. This function takes the following parameters:

Filename: is the name of the input file. The file should include only the sequences separated by a separator (the separator is'\n' but it can be changed). If you have a fasta file, please convert it to the right format using 'converter' (see below).

Output: is the type of output. We have 3 options:
'0': no output
'1': maximum overlap. This option uses two dimensional array. Accordingly, there is a limitation in term of space with this option.
'2': all overlaps are shown

Threads: number of threads to be used.

Min: is the minimum length of an overlap.

```
void AAPSP_PigeonHole_Approximate_Match(char *filename, char *output, int threads, int min, int mismatches,bool hamming);
```

This function finds the approximate maximum overlap or all approximate overlaps between every ordered pair of sequences. It takes the following parameters:

- Filename: the name of the input file.
- Output: '0' : no input
  '1' : maximum approximate overlaps. Since it uses two dimensional array, there is a limitation in term of space.
  '2' : all approximate overlaps are shown
  '3' : all approximate overlaps are shown and alignments are also shown. It only works with edit distance.
- Threads: number of threads to be used.
- Min: The minimum length of an overlap
- Mismatches the threshold of mismatches.
- Hamming: true : hamming distance is used
  False: edit distance is used

Additional auxiliary tools

- **Aof.cpp**: This file is for testing our library.
  To see how it works, run: Aof (with no arguments)
  Example: To run with 4 threads, output all overlaps, minimal length=30 and h=0 (exact matching):

  ```
  Aof test.txt -p 4 -o 2 -m 30 -h 0
  ```

- **gen_test.cpp**: You can generate random cases to test the code using this application. The program 'gen' will generate random strings. The user specifies 3 parameters:
  1- K (number of strings)
  2- N (total length of all strings)
  3- If the generated strings have equal sizes or not.
  To run: ./gen

- **check_a_string.cpp**: this application is used to retrieve a specific read. This app can be used to check correctness. To retrieve read 5:

  ```
  Example: to get read 5:
  check_a_string test.txt 5
  ```

  Please note that counting is starting from read 0.

- **converter.cpp**: This program is to convert a fasta file to the right input file. Example: converter text1.fasta text1.txt. if you have a fasta file, please use the converter first.
- **reverse_complement.cpp**: This program creates a reverse complement for every read in the input file. Accordingly, the size of the output file is double the size of the input file.

File Format

The input file should include the sequences only with a separator between every two consecutive sequences. We set '\n' as a separator, but it can be changed from tools.h
If you have a fasta file, please use the program 'converter' to convert a fasta file to an input file with the right format. DON'T USE FASTA FILE DIRECTLY.

- If you have any problem, please contact us:

Maan Haj Rachid
Qatar University
mh1108047@qu.edu.qa