alfy, Version 1.5: ALignment-Free local homologY

Mirjana Domazet-Lošo and Bernhard Haubold

June 29, 2012

1 Introduction

alfy is a program for comparing one or more query sequences to a set of subject sequences and determining the closest homologue among the subjects along each query [1]. In this document we describe how to get started with alfy and what input format it accepts. This is followed by a detailed Tutorial demonstrating central aspects of alfy when applied to HIV-1 genomes.

2 Getting Started

alfy was written in C and is intended to work on any UNIX system with a C compiler. However, please contact MDL at Mirjana.Domazet-Loso@fer.hr if you have any problems with the program.

- Unpack alfy

  ```
  tar -xvzf alfy_XXX.tgz
  ```

  where XXX indicates the version.

- Change into the newly created directory

  ```
  cd Alfy_XXX
  ```

  and list its contents

  ```
  ls
  ```

- Generate alfy

  ```
  make
  ```

  This creates two executables, alfy and alfy64. alfy is the 32 bit version of the program, alfy64 its 64 bit version. The 32 bit version can analyze up to $2^{29} \approx 5 \times 10^8$ bp, while the 64 bit version can theoretically analyze up to $2^{61} \approx 2 \times 10^{18}$ bp. In practice, the program is only limited by the available computer memory. On the other hand, the 32 bit version is somewhat faster than its 64 bit sibling and uses only half as much memory. We therefore recommend that you use the 32 bit version for analyzes of up to 5×10^8 bp.

- List options

  ```
  ./alfy -h
  ```

3 Input Files

Input sequences need to be in FASTA format and can contain only characters from the set $\{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$. The script cleanSeq.awk converts sequences to upper case and to removes all characters $\notin \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$ from the sequences (but not from their headers):

```
awk -f Scripts/cleanSeq.awk foo.fasta > foo2.fasta
```

4 Tutorial

- Annotate the query strain A+DQ083238 using 42 pure strains as subject:

```
./alfy -i Data/A+DQ083238.fasta -j Data/hiv42.fasta
>A+DQ083238
1 225 8.620536 A1+AF004885
226 495 25.146444 A1+AF069670
496 930 37.790272 G+AF084936
931 1155 6.579832 A1+AF069670
1156 1770 21.300528 C+U52953
1771 1875 16.382023 C+AF067155
1876 2175 11.652174 A1+AF069670
2176 2655 25.058455 A1+AF004885
2656 3285 23.612083 C+AF067155
3286 3585 29.481606 C+AY772699
3586 3825 14.728033 C+AF067155
3826 3915 15.022472 C+U52953
3916 4185 16.133829 C+AY772699
4186 4335 12.577181 C+U52953
4336 5025 23.910015 A1+AF004885
5026 5205 24.882681 G+AF061642
5206 5295 24.943821 A1+AF004885
5296 5415 10.008404 A1+AF069670
5416 5820 12.434650 A1+U51190
5821 6030 16.382023 A1+AF484509
6031 6225 8.395973 A1+AF069670
6226 6555 11.592705 A1+AF004885
6556 6705 7.747899 A2+AF286238
6706 6825 0.000000 A1+AF004885
6826 7140 11.314382 A1+AF484509
7141 7485 8.079027 A1+AF004885
7486 7755 18.382900 D+K03454
7756 8640 19.603571 A1+AF004885
8641 8805 1.378151 A1+U51190
8806 9330 8.914405 A1+AF004885
9331 9699 15.767802 A1+U51190
```

  This means that positions 1–225 of A+DQ083238 are most closely related to strain A1+AF004885. Further, across that interval the average shustring length of the shustrings induced by the "winning" subject sequence—A1+AF004885 in this case—is 8.6, which is a measure of the strength of the homology signal across that interval.

- To get a clearer view of the distribution of the various annotations, we can summarize the output:

```
./alfy -i Data/A+DQ083238.fasta -j Data/hiv42.fasta  |
awk -f Scripts/quantifyGenotypes.awk
>A+DQ083238
A1+AF004885 38.0
A1+AF069670 11.4
C+AF067155 10.1
A1+U51190 9.7
C+U52953 8.8
C+AY772699 5.9
A1+AF484509 5.4
```

```
G+AF084936 4.5
D+K03454 2.8
G+AF061642 1.9
A2+AF286238 1.5
```

This indicates that 38.0% of strain A+DQ083238 is most closely related to strain A1+AF004885, 11.4% to strain C+AF067155, etc. We can further summarize the result by collapsing annotations for strains that belong to the same group:

```
./alfy -i Data/A+DQ083238.fasta -j Data/hiv42.fasta  |
perl -pe 's/(.)\+.+?(\s)/$1$2/g' |
awk -f Scripts/quantifyGenotypes.awk
>A
A1 64.6
C 24.7
G 6.3
D 2.8
A2 1.5
```

- alfy works in three steps:
  1. Construction of the enhanced suffix array from all query and subject sequences;
  2. determination of "winning" subjects across contiguous intervals along a query;
  3. sliding window analysis of these closest neighbors intervals to find the final annotation of a given query.

  The last step is sensitive to the length of the sliding window. By default this is set to 300 bp; if we set it to 400 bp, the results change somewhat, but not dramatically so:

```
./alfy -w 400 -i Data/A+DQ083238.fasta -j Data/hiv42.fasta  |
perl -pe 's/(.)\+.+?(\s)/$1$2/g' |
awk -f Scripts/quantifyGenotypes.awk
>A
A1 63.1
C 26.8
G 7.6
D 2.5
```

- We may only be interested in "strong" homology signals in our analysis. These are defined as regions where the average shustring length is greater than the maximum shustring length occurring by chance alone. Regions with an average shustring length below the threshold value are marked `nh` for "no homology":

```
./alfy -i Data/A+DQ083238.fasta -j Data/hiv42.fasta  -M
>A+DQ083238
1 225 8.620536 A1+AF004885
226 495 25.146444 A1+AF069670
496 930 37.790272 G+AF084936
931 1155 6.579832 nh
1156 1770 21.300528 C+U52953
1771 1875 16.382023 C+AF067155
1876 2175 11.652174 A1+AF069670
2176 2655 25.058455 A1+AF004885
2656 3285 23.612083 C+AF067155
3286 3585 29.481606 C+AY772699
3586 3825 14.728033 C+AF067155
3826 3915 15.022472 C+U52953
```

```
3916 4185 16.133829 C+AY772699
4186 4335 12.577181 C+U52953
4336 5025 23.910015 A1+AF004885
5026 5205 24.882681 G+AF061642
5206 5295 24.943821 A1+AF004885
5296 5415 9.539326 A1+AF069670
5416 5820 13.280936 A1+U51190
5821 6030 16.382023 A1+AF484509
6031 6225 8.395973 A1+AF069670
6226 6555 11.592705 A1+AF004885
6556 6705 7.747899 nh
6706 6825 0.000000 nh
6826 7125 11.475837 A1+AF484509
7126 7275 2.210084 nh
7276 7485 11.358851 A1+AF004885
7486 7755 18.382900 D+K03454
7756 8505 20.278667 A1+AF004885
8506 8640 11.820225 A1+AF004885
8641 8835 1.378151 nh
8836 9330 9.064439 A1+AF004885
9331 9699 15.767802 A1+U51190
```

- As before, we can summarize the annotations:

  ```
  ./alfy -i Data/A+DQ083238.fasta -j Data/hiv42.fasta  -M |
  perl -pe 's/(.)\+.+?(\s)/$1$2/g' |
  awk -f Scripts/quantifyGenotypes.awk
  >A
  A1 57.5
  C 24.7
  nh 8.7
  G 6.3
  D 2.8
  ```

  We find that the K annotation is now missing altogether and 8.7% of the annotations are revealed as "weak" (nh).

- To further investigate what a nh annotation means, we can extract such a region and blast it against the subject sequence:

- The amount of weakly homologous regions is sensitive to the minimal length of a recombining fragment. This quantity is set by the -f option with a default value of $-\mathrm{w}/4 = 75$. Increasing this value leads to fewer regions diagnosed as "non homologous". For example, if we double the value of -f we get

  ```
  ./alfy -f 150 -i Data/A+DQ083238.fasta -j Data/hiv42.fasta  -M |
  perl -pe 's/(.)\+.+?(\s)/$1$2/g' |
  awk -f Scripts/quantifyGenotypes.awk
  >A
  A1 64.0
  C 24.7
  G 8.5
  D 2.8
  ```

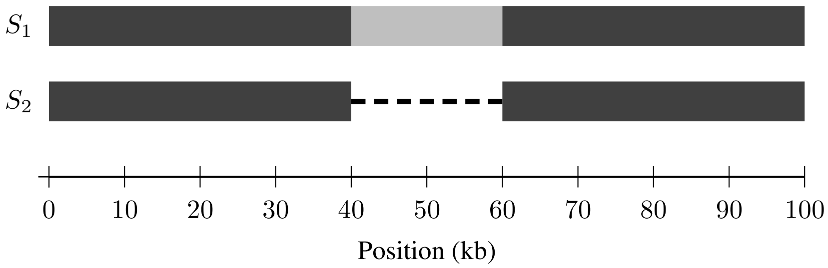

Figure 1: Homology structure of the two simulated sequences S_1 and S_2. *Dark gray*: homologous regions; *light gray*: sequence without homology; *dashed line*: gap.

- The previous result is again sensitive to the parameter settings. In this case the central quantity is the significance value for the maximum shustring length. The mathematical theory for this was derived in [2] and it is set with the `-P` option. This can be interpreted like a classical P-value: it is the error probability when rejecting the null hypothesis that a shustring of some given length is due to chance alone. The default value of the `-P` option is 0.4, which may strike you as rather high—why not use the classical $P = 0.05$ threshold? It is important to realize that a query has to contain not just one shustring that is longer than the maximum shustring length, but this property has to apply on average to *all* shustrings across a given window. If we set `-P` to a much more stringent value like 0.05, we get

  ```
  ./alfy -i Data/A+DQ083238.fasta -j Data/hiv42.fasta  -M -P 0.05 |
  perl -pe 's/(.)\+.+?(\s)/$1$2/g' |
  awk -f Scripts/quantifyGenotypes.awk
  >A
  >A
  A1 40.3
  nh 28.3
  C 23.2
  G 5.3
  D 2.9
  ```

  The false negative rate is now misleadingly high (28.3%).

- Next, we explore the opposite problem of an inflated false positive rate. The files `s1.fasta` and `s2.fasta` contain the two simulated homologous sequences S_1 and S_2, respectively. The sequences are separated by 100 mismatches per kb and have the homology structure shown in Figure 1: S_2 contains a gap between positions 40–60 kb, which is duly detected by `alfy` as a region without homology when running it with S_1 as query and S_2 as subject:

  ```
  ./alfy -i Data/s1.fasta -j Data/s2.fasta -M
  >S1
  1 40125 13.227115 S2
  40126 60045 9.835734 nh
  60046 100000 13.444236 S2
  ```

  However, if we set `-P` from its default value of 0.4 to, say, 0.5, a semblance of homology across the 20 kb gap is created:

  ```
  ./alfy -i Data/s1.fasta -j Data/s2.fasta  -M -P 0.5
  >S1
  1 100000 12.639876 S2
  ```

- Theoretically, $P = 0.5$ should result in exact equality between the threshold length and the average length of random shustrings. Since the theory strictly applies only in the limit of infinite sequence length [2], we observe a slight deviation from this ideal, but $P = 0.45$ will do the trick:

```
./alfy -i Data/s1.fasta -j Data/s2.fasta -M -P 0.45
>S1
1 40125 13.227115 S2
40126 60045 9.835734 nh
60046 100000 13.444236 S2
```

- If you would like to simulate your own sequences to explore the behavior of `alfy`, here is the protocol for generating `s1.fasta` and `s2.fasta`:
  - Create two homologous 100 kb sequences separated by 100 mismatches per kb

    ```
    ms 2 1 -s 1000 -r 0 100000 | ms2dna > tmp.fasta
    ```

    The program `ms` is the coalescent simulator written by Dick Hudson [3], which is available from his web site. The program `ms2dna` and the two programs `getSeq` and `cutSeq` applied in the next steps are part of the `bioBox` collection of tools available from

    `http://guanine.evolbio.mpg.de/bioBox/`

  - Extract `s1.fasta`

    ```
    getSeq -s S1 tmp.fasta > s1.fasta
    ```

  - Extract `s2.fasta` and make the deletion

    ```
    getSeq -s S2 tmp.fa |
    cutSeq -s -r 1-40000,60001-100000 > s2.fasta
    ```

5 Change Log

1. Version 1.1 (February 4, 2011)
   - First stable version.
2. Version 1.2 (February 6, 2011)
   - Changed default value and the interpretation of `-P`.
   - Changed bracketed output that indicates "no homology" to `nh`.
   - Changed positions as starting from 0 to starting from 1.
   - Worked on documentation.
3. Version 1.3 (March 11, 2011)
   - Improved documentation.
4. Version 1.4 (December 8, 2011)
   - Fixed crash caused by contigs that are shorter than the default window length (`-w`).
   - Increased default value of minimum recombining fragment (`-f`).
5. Version 1.5 (June 29, 2012)
   - The linker didn't accept `-lm` at the beginning of the library list in `Src/Alfy/Makefile`, so moved it to the end of the list.

6 Acknowledgement

This software is based on the `dss_sort` library by G. Manzini [4].

References

[1] M. Domazet-Lošo and B. Haubold. Alignment-free detection of local similarity among viral and bacterial genomes. *Bioinformatics*, 27:1466–1472, 2011.

[2] B. Haubold, P. Pfaffelhuber, M. Domazet-Lošo, and T. Wiehe. Estimating mutation distances from unaligned genomes. *Journal of Computational Biology*, 16:1487–1500, 2009.

[3] R. R. Hudson. Generating samples under a Wright-Fisher neutral model of genetic variation. *Bioinformatics*, 18:337–338, 2002.

[4] G. Manzini and P. Ferragina. Engineering a lightweight suffix array construction algorithm. In *ESA '02: Proceedings of the 10th Annual European Symposium on Algorithms*, pages 698–710, London, UK, 2002. Springer-Verlag.