Spaln の利用法　(2018/7/26 最新更新)

1. インストール

以下で、*work* は作業用ディレクトリを、*seqdb* はゲノム配列やアミノ酸配列データベースを保存するディレクトリを、*download* はインストール用の作業ディレクトリを表す。

I. 実行型を利用し、一つの作業ディレクトリーに全てをコピーする場合
a) % mkdir *work* (完全パス名)
b) % cd *work*
c) % spalnXX.PC.tar.gz （実行型）をダウンロード
(XX: version code, PC: platform code).
d) % gzip –cd spaln.tar.gz | tar xf –
e) *work* /bin をパスに加えるか、*work* /bin の中身をパスの通ったディレクトリ（/usr/local/bin など）にコピーまたは移動
f) % mv ./table/* .; rmdir ./table
g) % mv ./seqdb/* .; rmdir ./seqdb
h) ２に進む

II. *work* が *seqdb* またはテーブルディレクトリと異なる場合
a) % mkdir *download* (完全パス名)
b) % cd *download*
c) % spalnXX.PC.tar.gz（実行型）をダウンロード
(XX: version code, PC: platform code).
d) % gzip –cd spaln.tar.gz | tar xf –
e) *download*/bin をパスに加えるか、*download*/bin の中身をパスの通ったディレクトリにコピーまたは移動
f) % setenv ALN_TAB *download*/table (csh/tsh) or
g) $ export ALN_TAB=*download*/table (sh/bsh)
h) % setenv ALN_DBS *download*/seqdb (csh/tsh) or
i) $ export ALN_DBS=*download*/seqdb (sh/bsh)
j) f)-i)内の２行をシェルに応じて.bashrc などの rc ファイルに書き込んでおくとログインごとに入力する手間が省ける
k) ２に進む

III. 標準設定でコンパイルする場合
a) % mkdir *download*
b) % cd *download*
c) spalnXX.tar.gz（ソースコード）をダウンロード
d) % gzip –cd spaln.tar.gz | tar xf –
e) % cd spalnXX/src（XX はバージョン番号）
f) % ./configure
// もし Makefile の中の$(CC)が C++コンパイラーでなければ Makefile を適切に編集すること（例えば、CC = g++）。C コンパイラー（gcc など）ではコンパイルエラーになる。
g) % make
h) % make install
// 実行形式のプログラムは ../bin　にコピーされる

// ../table にアミノ酸置換行列のファイルが書き込まれる

i) *download*/spalnXX/bin を PATH に加える。またはその中身をパスの通ったディレクトリにコピーまたは移動

j) `% make clearall` // オブジェクトファイルの消去

IV. 標準設定以外のディレクトリにインストールする場合は
III-f の段階で
`f') % ./configure --help`
を入力すれば設定法が表示される。

2. 配列データ

a) % cd *seqdb*

b) ゲノム配列またはアミノ酸配列データベースの配列をこのディレクトリにコピーまたはリンクする。配列は multi-fasta 形式でなければならない。

c) 染色体に分かれている場合は、cat でひとつにまとめる。まとめたものを仮に xxxgnm.mfa とする。以下の make の操作を有効にするために、エクステンションを. mfa または.gf とする。
xxxgnm.mfa 内の'>'で始まる行は
>GgggssssC… comments
という形が望ましいが、fasta 形式に従う限り必須ではない。Gggg の部分が属名の初めの 4 文字、ssss が生物種の初めの 4 文字を表す（4 文字より少ない場合は_で補う、例 : Arabthal、Mus_musc）。C…の部分が染色体、scaffold、または contig を表す。この ID 名は各ファイル内で重複しないように。
アミノ酸配列データベースの場合には、ファイル名のエクステンションを.faa とする。以下、ファイル名を prosdb.faa と仮定する。各配列の ID はユニークであれば他に特別な制約はない。ソートされていなくてもよい。

d) Version 2.3.2 以降、gzip で圧縮したファイル（e.g. xxxgnm.mfa.gz）を直接入力とすることができるようになった。

e) % makeidx.pl -in xxxgnm.mfa(.gz) // 塩基配列 query 用に index を付ける。
% makeidx.pl -ip xxxgnm.mfa(.gz) // アミノ酸配列 query 用に index を付ける。
% makeidx.pl -ia prosdb.faa(.gz) // アミノ酸配列データベースの場合。
- この計算に 10 数分かかることがある。

f) プラットフォームによっては、ひとつのファイルがあるサイズ以上になると上の操作が失敗するかもしれない。そのときは cat でひとつにまとめず、複数の染色体配列を直接 makdbs や spaln の引数に与えることによって問題を回避できる場合がある。詳しくは英文のマニュアルを参照のこと。

g) 上の操作で *xxxgnm.grp* または *prosdb.grp* ファイルが作成されていれば、以下のオプション値は makidx.pl スクリプトから呼び出される makblk.pl によって自動計算される。
＜注意＞ ゲノム配列の一部（例えば染色体 1 本）だけをフォーマットすると以下の最大遺伝子長の推定値が短すぎる場合がある。その場合、-XG*N* オプションを makblk.pl の引数に付けるか、spaln の実行時に付けるかして適切な予想最大遺伝子長を指定する必要がある。*N*の後に k または M を付けると、期待通りキロまたはメガ塩基の意味になる。
各オプションの意味：() 内ディフォールト値
1. –Xk*N* *N*は単語長（11 for DNA 5 for Protein）
2. –XG*N* *N*は最大遺伝子長（262144）
3. –Xb*N* *N*はブロック長（4096) // N < 65536、(全ゲノム長／N) < 65536

でなければならない。*N*の目安は sqrt（全ゲノム長）。哺乳動物では *N* ≒ 54000

4. –Xa*N*　期待値よりも *N* 倍以上多く存在する単語を無視する（一種の repeat mask）（10）。
5. –Xs*N*　ブロック内シード間距離（=*k*）。*N*(1≤*N*≤*k*) を小さくするとブロック検索の感度が高くなる。反面、単語表が *k*/*N* 倍大きくなる。問い合わせ配列長が短い場合に特に有効。

- この計算には 30 分ほどかかることがある。

3. 実行

a) % cd *work*
b) (multi-)fasta の cDNA 、アミノ酸、またはゲノム断片配列を準備（仮に *query* とする）
c) % spaln –Q[0|1|2|3] [–O*N*] [-M*N*] [-o*Output*] [–T*xxx*] genome_fragment *query*
d) % spaln –Q[4|5|6|7] [–O*N*] [-M*N*] [-o*Output*] [–T*xxx*] –d*xxxgnm query*
e) % spaln –Q[4|5|6|7] [–O*N*] [-M*N*] [-o*Output*] [–T*xxx*] –a*prosdb query*
f) % spaln –Q[4|5|6|7] [–O*N*] [-M*N*] [-o*Output*] [–T*xxx*] *xxxgnm.mfa query*
g) % spaln –Q[4|5|6|7] [–O*N*] [-M*N*] [-o*Output*] [–T*xxx*] *prosdb.faa query*

- -Q0,4 では HSP 探索を行わずに直接 DP 計算を実行する。他と比べて遅い。-Q1,5 では、1 レベルの HSP 探索を行う。-Q2,6 では、レベル 1 で HSP が見つからなかった範囲に対して、より短いシードを用いてレベル 2 の HSP 探索を行う。-Q3,7 ではこの再帰をさらにもう一度くり返す。
- d）の形式では、*query* が核酸またはアミノ酸配列かによって、適切なインデックスが選択される。
- e）の形式では、*query* 中の一定長以上の ORF ごとに *prosdb* に対する類似性検索を行う。最も類似性が高かった配列を鋳型としたスプライスアラインメントにより *query* 中の遺伝子を予測する。
- f), g)形式では、フォーマット（index 付け）と検索を同時に行う。しかし、作成された index は一度使われるだけで、ファイルとして保存されない。g)形式は、blastp のような検索を手軽に行うのに適している。ただし、blastp と異なり、大域的なアラインメントを計算する。
- -O*N* option
  1. –O0　GFF3 gene format
  2. –O0 g)　大域的アラインメントの統計量（一致度など）
  3. –O1　アラインメント
  4. –O2　GFF3 match format
  5. –O2 g)　Sugar format
  6. –O3　BED/psl format
  7. –O4　exon-oriented (megablast –D 3 like)な出力形式
  8. –O4 g)　xy-座標+ギャップなしアラインメント長形式
  9. –O5　intron-oriented な出力形式
  10. –O6　エキソンを結合した”cDNA”配列
  11. –O7　翻訳アミノ酸配列
  12. –O8　Cigar form
  13. –O9　Vulgar form
  14. –O10　SAM form
  15. –O11　BAM form (予定)
  16. –O12　-O4 の形式をバイナリーファイルに書き出す。もし、-o*Output* を指定すれば、Output.grd, Output.erd および Output.qrd というファイルが、していなければ、xxx.grd, xxx.erd および xxx.qrd というファイル

が作成される。

17. –O15　Query アミノ酸配列のエキソン構造を推定。database 中の最も近縁のゲノム配列とのアラインメントに基づく。

- -M*N* option
  1 query あたりの最大出力数。-M オプションを付けないとき（*N*=0）は、1。*N*を指定しないときはプログラムの指定値（*N*=4）。*N*=1 のときは、一回目でアラインメントできなかった領域を再検索。
- –T*xxx* の *xxx* の部分は~/table/内の *xxx* サブディレクトリーに対応する。Version2.2.2 からは、gggssss 形式の生物種、あるいは属名を用いることも可能になった。（例えば、-Tarabthal. –TArabidopsis, -TEudicoty は同じパラメータセットを指定する。）そのためには~/table/内に新しい形式の gnm2tab ファイルが必要である。対応するものがないときには一番近そうなものをユーザが選ぶこと。現時点では 102 のパラメータセットが用意されている。これらの多くは、いくつかの生物種の混合から推定されたものである。DNA を問い合わせ配列として種特異的なパラメータを用いる場合、-yS または-yX オプションが必要であるが、アミノ酸が問い合わせ配列の場合は不要（このオプションは常に設定されている）。
- *query* の部分を *'query (from to)'* とすれば、from 番目のエントリーから to 番目までのエントリーについてのみ計算する。複数の CPU で計算するときには、
  % spaln –Q5 –O12 -oxxxO1 –dxxxgnm *'query* (1 1000)'
  % spaln –Q5 –O12 -oxxxO2 –dxxxgnm *'query* (1001 2000)'
  などとすればよい。結果は付随の sortgrcd コマンドを用いて統合する。
  しかし、マルチスレッド環境では上の操作はほぼ不要。
- その他のコマンドラインオプション（代表のみ）
  -H*N*　全体スコアの閾値
  -LS　Smith-Waterman 型の両末端処理
  -ia　入力ファイル中にゲノム配列と問い合わせ配列を交互に配置（Q<4）
  -ia　一つのファイルにペアドエンド配列を交互に配置（Q≥4）
  -ip　第 1 引数のゲノム配列と第 2 引数の問い合わせ配列を同期比較（Q<4）
  -ip　二つのファイルにペアドエンド配列を同じ順に配置（Q≥4）
  -pa　polyA 配列を除かない
  -pq　標準エラーへの出力を抑制
  -pw　閾値以下のスコアをもつ結果も出力
  -t[*N*]　CPU 数 *N*で並列計算。*N*を省略すると全ての CPU を使う
  -u*N*　ギャップ伸長ペナルティ（3, 2, 2）
  -v*N*　ギャップ開始ペナルティ（8, 6, 9）
  -ya*N*　正規な（GT..AG, GC..AG, AT..AC）境界以外も候補として考慮
  -yl3　ダブルアフィンギャップペナルティ
  -ym*N*　塩基の一致に対するスコア（2, 2）
  -yn*N*　塩基の不一致に対するペナルティ（6, 2）
  -yL*N*　最短イントロン長（30, 30）
  -yS　-yS100 と同等。種特異的なパラメータを利用。アミノ酸配列が問い合わせのときは、第一フェーズで「サルベージ」を行うことを表し、別の意味になる
  -yS*N*　種特異的なイントロン境界シグナルの寄与率を *N*%に設定。*N*=0 では、イントロン末端の 4 塩基だけに依存した普遍シグナルのみを用いる。
  -yX　問い合わせが DNA かアミノ酸かで意味が逆転。DNA では種間比較用

のパラメータに設定。以下括弧内の２番目の数値。-yS100 オプションも同時に自動設定。逆にアミノ酸配列では種内比較用に設定。
その他については英文マニュアルを参照のこと。

4. 実行結果の整理

Sortgrcd、染色体上の位置による整列、統合とフィルタリング
% sortgrcd [options] xxx*.grd

- Options:
  1. –C*N*　最小カバー率（0-100）
  2. –P*N*　最小一致率（0-100）
  3. –H*N*　最小アラインメントスコア
  4. –m*N*　エキソン境界 20 塩基中の最大ミスマッチ数
  5. –u*N*　エキソン境界 20 塩基中の最大ギャップ内塩基数
  6. –n1　非正規な（GT..AG, GC..AG, AT..AC 以外）境界を許す
  7. –V*N*　コアソートに用いる内部メモリーのバイト数。データサイズがこれより大きければ、何回かに分けてソートする。*N* にサフィックス k または K を付ければキロバイト単位、m または M を付ければメガバイトであるとみなす（例、-V10M）。
  8. –Sa　染色体、コンティグ識別子の辞書順
  9. –Sb　ヒット数の多い染色体の順番に出力
  10. –Sc　元のゲノム配列上の並び順に出力（標準）
  11. –Sr　マイナス鎖の場合、後方のものから順に出力する
  12. –O*N*　出力フォーマット。*N*=0,3 - 7 は spaln のものと同じ。–O15 では重複を除きユニークはイントロンのみを–O5 形式で出力する。
- ディフォールトでは上記フィルターを掛けない。
- –C, P, H オプションは転写産物単位で、–m, n, u オプションはエキソンまたはイントロン単位でフィルタリングする。
- spaln の出力が複数のファイルに分かれているときにも、ひとまとめとしてソートする。引数には*grd だけを指定するが、対応する*.erd および.qrd ファイルも同じディレクトリに存在しなくてはならない。
- エキソン領域が 1 塩基でも重なれば同一 locus とみなす。Locus の区切りは!で始まる行。

5. 例題

配布の seqdb ディレクトリ内で

- make ddignm.idx
- make ddignm.bkn
- make ddi.cdna
- make ddi.srd

と入力すれば、上記一連の作業を実行する。

6. 注意

- 異なるアーキテクチュアの計算機（例えば、32 ビットと 64 ビットの Linux）でフォーマットしたデータベースファイルは共有できない。
- バージョン 1.3 からアミノ酸配列を問い合わせとすることが可能となった。
- バージョン 1.4 からゲノム配列断片を問い合わせとすることが可能となった。この場合、-a オプションによってアミノ酸配列データベースを指定する。
- バイナリー配列データ（.idx, .bkn など）のフォーマットが何度か変更された。下位のバージョンは上位のバージョンでフォーマットされたファイルと整合しないので再フォーマットが必要。同一バージョンでフォーマットされたファイルを用いることが望ましい。

- アミノ酸置換行列（mdm）の形式も変更されたので、新たに makmdm を走らせる必要がある。
- バージョン 2.0 からは C++（g++）のみでコンパイル可能となった。C（gcc）は使えないので注意。
- 一方、マルチスレッドによる並列処理が可能となった（-t オプション）。
- また、以前より多くの種類の特徴量を用いることが可能となり、進化的に遠い配列間のアラインメント精度が向上した。
- バージョン 2.1 から配列データベースのフォーマットが変更された。以前のバージョンでフォーマットされたデータベースを使用可能だが、逆は不可。これに伴い、付属プログラムの makdbs および sortgrcd もバージョン 2.0 に更新された。
- 上の変更によって、対象および問い合わせ配列識別子の文字数に関する制限が取り除かれた。
- sortgrcd の標準の出力順が変更された。
- バージョン 2.2 から、アミノ酸配列を問い合わせとする、アミノ酸配列データベースの検索が可能となった。Blastp と異なり、局所アラインメントではなく大域的なアラインメントを計算する。Blastp より一ケタほど高速であるが、弱い類似性を検出能力は劣る。
- また、フォーマットと検索を同時に行うオプションも追加された。ただし、フォーマットの結果は保存されない。
- バージョン 2.3.2 から gzip で圧縮されたファイル（X.mfa.gz, X.gf.gz, X.faa.gz, X.seq,gz, X.bka.gz, X.bkn.gz, X.bkp.gz, X.grd.gz, X,erd.gz, X.qrd.gz）を入力とすることが可能となった（ただし、X,idx, X,ent, X.grp, X.odr を圧縮してはならない。）そのためにはコンパイルの段階で、./configure –use_zlib=1 を指定する必要がある。Zlib がインストールされていないシステムでは新たに入手する必要がある。すべての zlib のバージョンをテストしたわけではないので、コンパイルが失敗する可能性がある。