PALS User Guide

Version 1.0
January 2005

Software written by Robert C. Edgar and Eugene W. Myers.

Manual written by Robert C. Edgar.

http://www.drive5.com/pals

Please visit the web site for requested citation, updates to the software and for information on how to contact the authors for help and feedback.

Introduction

PALS stands for Pairwise Aligner for Long Sequences. In its current 32-bit implementation, PALS finds local alignments between DNA sequences of up to a few hundred megabases. PALS does not output information required to align each base; rather, it outputs the coordinates of the beginning and end of each hit, an alignment score, and a lower bound on the identity of the alignment. So if you need to know where every gap must be placed to make the alignment explicit, you will need to pass the hits produced by PALS to another alignment program. (This feature is not trivial to add to PALS, and would result in longer execution times.)

Quick start

Following is a quick summary of how to use PALS. This may be all you need to know.

Input files must be DNA sequences in FASTA format. There may be more than one sequence in an input file. The letters AGCT stand for nucleotides, N (and any other letter except ACGT) means "unknown nucleotide".

Output is in GFF format (see *Output File Format* below).

To align two different sets of sequences:

```
      pals -target t.fasta -query q.fasta -out t_q_hits.gff
```

To align a set of sequences to itself:

```
      pals -self t.fasta -out t_t_hits.gff
```

The minimum hit length is specified by the -*length* option, the minimum identity by the -*pctid* option:

```
      pals -self t.fasta -out t_t_hits.gff -length 100 -pctid 98.0
```

Defaults are minimum length 400 bases, minimum identity 94.0%.

The algorithm

PALS was originally designed for use in PILER, an algorithm that identifies an classifies repeated elements in genomes. PALS is well suited for use in PILER for several reasons. PILER does not need individual gaps, so computational resources are saved by having a compact representation of hits. Also, PALS has optimizations for aligning a sequence to itself, and reports all hits to a given region of a target sequence. Compare with alternative aligners that are optimized for other applications, such as database search, and may restrict the number of hits to a given location. Also, PALS maintains relatively good performance on highly repetitive unmasked DNA.

PALS searches for all hits that are (a) at least a given length, and (b) are at least a given identity. By default, PILER reports all hits that are = 400 bases and have = 94% identity. In general, these search criteria yields a distinctively different set of hits compared with aligners that use an *e*-value threshold (which will tend to find shorter alignments with higher identity and longer alignments with lower identity). A technical advantage of the PILER method compared with some other aligners is that very little time is wasted on alignments that do not meet the search criteria, in contrast to *e*-value aligners that must compute an alignment in order to determine whether its score falls below the threshold.

The algorithm has two phases. The first phase (the *filter*) narrows the search space. The output from the filter is a set of seed hits which is guaranteed to contain all hits that meet the requested criteria, but may contain false positives. The second phase (*dynamic programming*, or *DP*) examines the search space to find the endpoints and scores of local alignments meeting the given criteria. The core algorithms are due to Gene Myers, Jens Stoye and Kim Rasmussen. The banded search variant was developed by Bob Edgar. The algorithms are not published at the time of writing. The method is very fast and sensitive for certain applications, such as PILER.

Banded search

A *banded* search is a special case of self-alignment that looks for similarity between regions that are close together in the sequence. The maximum distance between regions in a single alignment is the *diameter* of the search, which is measured from the start (or end) of one region to the start (or end) of the other region.

Parameters

Usually, alignment parameters are specified by -*length* and -*pctid*, the minimum length and identity of a hit. Expert users can specify filter and DP parameters. A full explanation requires a specification of the algorithm, which is currently unpublished and will not be described here. Contact the authors if you are interested in the details.

From the given length and identity parameters, PALS attempts to determine appropriate parameters for the filter and DP stages, given the amount of available memory. This may fail if the length is too short, if the identity is too low, if the input sequence(s) are too long, or if there is too little memory. The process of searching for suitable parameters is traced to the log file, which is useful when the search fails. Parameters are set using one the following sets of options.

Option set	**Description**
(none)	Default. Equivalent to -*length* 400 -*pctid* 94.
-*length* -*pctid*	PALS attempts to determine filter parameters from the given length and identity. This may fail.
-*wordsize* -*seedlength* -*seeddiffs* -*length* -*pctid*	Specifies all filter and DP parameters. Tube offset is set to 32 by default, may optionally be specified by -*tube*.

Output file format

GFF files are text files with one record per line. At the time of writing, a specification can be found here:

http://www.sanger.ac.uk/Software/formats/GFF/

Fields are separated by tabs. A typical output record is:

```
chr3  pals  hit  10863665  10864065  310  +  .  Target  chr1  17255  17657;  maxe  0.058
```

Fields are as follows.

Field number	**Description**	**Comments**
1	Query sequence name	PALS truncates the FASTA annotation at the first blank character.
2	Source	Is always set to *pals*.
3	Feature	Is always set to *hit*.
4	Start	Position within the sequence of hit start. The first base is at position 1 (GFF uses 1-based coordinates).
5	End	Position of hit end.
6	Score	Alignment score. Matches score +1, differences (substitutions, deletions and insertions) score –3.
7	Strand	Is + if the aligned regions are on the same strand, – if the two regions are reverse complemented with respect to each other.
8	Frame	Is always set to ".", a period.
9	Attributes	Formatted as follows: *Target name start end ; maxe e* Fields are separated by spaces (not tabs). *Name* is the name of the target sequence, *start* and *end* are the coordinates of the matching region, and *e* is an upper bound on the error ratio, i.e. the number of differences between the two regions divided by the length of the region. A difference is a substitution, deletion or insertion of length one. E.g., *e*=0.058 means identity = 94.2%.

Other options

-log *filename*

Write progress information to the given file. Useful for trouble-shooting.

-loga *filename*

As -log, except appends to an existing file instead of overwriting.

-maxmem *bytes*

To automatically determine filter parameters, PALS needs an upper bound on the amount of memory that can be used. On some operating systems, PALS can detect the amount of physical RAM installed. If so, the default upper bound is 80% of the installed RAM. If the amount of physical RAM is not known, 1 Gb is assumed. To change the upper bound, specify *-maxmem*. The number of bytes can be specified using integer, fixed point or scientific notation, e.g. *-maxmem 0.5e9* for 500 Mb.

-diameter *bases*

Maximum distance between regions in an alignment. By default, this is the length of the sequence. If a diameter that is much less than the sequence length is used, then generally higher sensitivity can be achieved (shorter lengths and lower identities are allowed), and / or the search is significantly faster and uses significantly less memory.