ParSeq Documentation

1 Which problems can be handled by ParSeq?

The program is able to search for the occurrence of strings on long character sequences that can be described by special regular expressions. An extended alphabet is available for these regular expressions, with which biochemical attributes can be described alongside character sequences.
Biological background:
Long DNA-Sequences, like bacteria genomes, parts of eucaryal genomes or the amino acid sequences of proteins can be understood as character sequences. These are to be scanned for the appearance of characteristic or functionally important areas. (DNA: promoters, protein binding sites. Amino acid sequences: functional groups etc.) Such sections are often marked by conserved sequences (consensus sequences). They can appear separately or as a pattern (as a series of motifs in a defined distance to each other). The program is able to search for predefined sequences (with and without allowed errors) as well as consecutions of multiple motifs where the distances of the motifs is variable.
If the DNA is translated into an amino acid sequence the resulting character sequences can furthermore be searched for part-sequences, which possess defined biochemical properties, to the extent to which they can be derived from the sequence-forming aminoacids (AA) (calculated using scaling tables). Thus, amino acid sequences can be searched for sections with a certain function too, as long as conditions (consensus sequences, areas with fixed biochemical attributes) can be specified whose fulfillment is with high probability attached to the existence of the biological function. (Search for reactive groups, signal sequences, etc.)

2 What input is required?

Input is needed in the form of the sequences that are to be scanned (text files) and the regular expression that is to be searched for.
The program offers the possibility to choose sequence files (text files) available over the file system of the local computer for a query. The program automatically saves the path of the file that was last chosen. The user can distinguish between DNA and protein files. Depending on which one is chosen, a different abbreviation table is used (see appendix) and in the case of protein files the result output includes a conversion to the DNA position. The query always refers to all selected files.
The path to the selected files is saved for further queries, in addition to that the intermediate results are stored in a directory structure where they can be searched trough afterwards. Thus, a rough search can be done first and then it can be improved step by step.

3 Syntax of regular expressions

The patterns that are to be searched are described by extended regular expressions. As a basis the syntax of existing libraries (java.util.regex, regex++, Perl ...) is used. It is supplemented by some additional possibilities The most simple components are character strings. Parts of expressions are divided by „@“.

You are searching for a string that ...	Example
...exactly matches the specified one	(ABC)*
...may contain any characters, the length of the string is characterized by minimum and maximum length	(X{min,max})*
...alternatively contains one of the specified characters	(A|B|C)
...contains the specified string as often as specified in „Number“	((ABC){Number})

* Parentheses can also be left out here.

Example:
(ABBA)@X{3,5}@(A|B|D)
Description:
You are searching for a definite string ABBA, followed by a character string with a length of 3 to 5 arbitrary characters, followed by A or B or D. Additionally, side conditions can be defined for every part of the expression (until the next @-symbol). Side conditions are biochemical properties or distances. The side conditions, have defined names and an argument list. They are appended by “/”. After “/” a list of the arguments follows, divided by semicolons. (For the list of names see appendix)

Example:
(ABC)@X{20}/hdp_kd (5,>,0)@
Description:
You are searching for a definite string ABC, followed by a character string with a length of 20 arbitrary characters (AA), where the character string is supposed to have a hydrophobicity score bigger than 0 according to Kyte/Doolittle which is measured as the average value in a 5-character-window. (The window size used for the averaging is selectable between 1 and 25. Apart from >, < and = are possible logical operators.)

Example:
(ABCDEFG)/hd(2)
Description:
You are searching for the stated character string where a Hamming distance of 2 is allowed.
The edit distance is implemented as additional error measurement, abbreviation ed.

4 Result output

The result output consists of several parts:

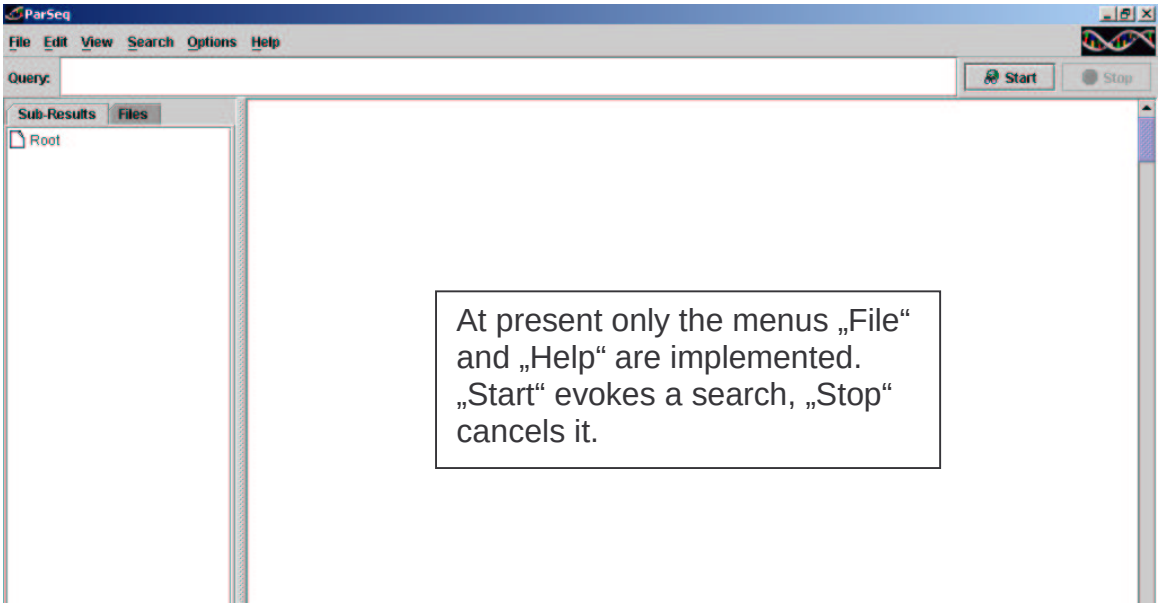
First, we get information about the accomplished query:
Primarily, the entered search expression is shown, then the regular expression (regex) that was calculated from it. That way, with one allowed error all possible versions are searched successively. These are listed. Additionally the program states which file has been scanned.

Then, the list of hits follows.
Stated is the hit sequence, the position on the scanned character string and in the case of amino acid sequences also the position of the hit on the DNA. Moreover, there is a hit indication with a marking of the hit's structure. (Beginning and end of the individual part of the expression)

At last a statistics follows:
Stated is how many files of what sizes have been scanned, the number of hits and the computing time.

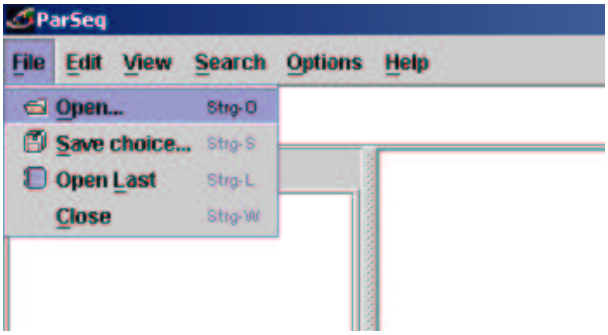
5 User interface

Screenshot



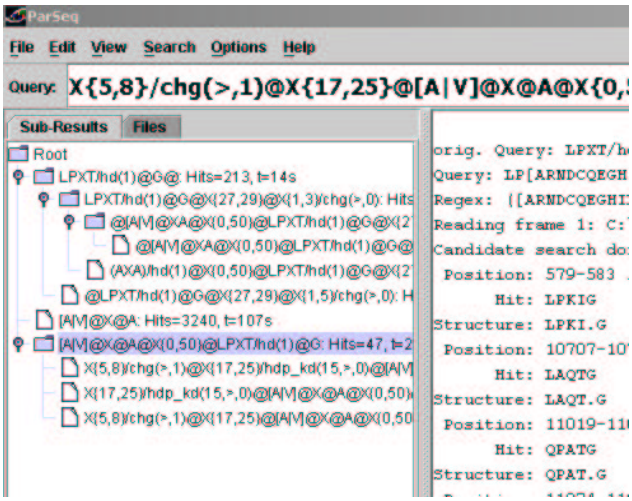
At present only the menus „File“ and „Help“ are implemented. „Start“ evokes a search, „Stop“ cancels it.

File selection



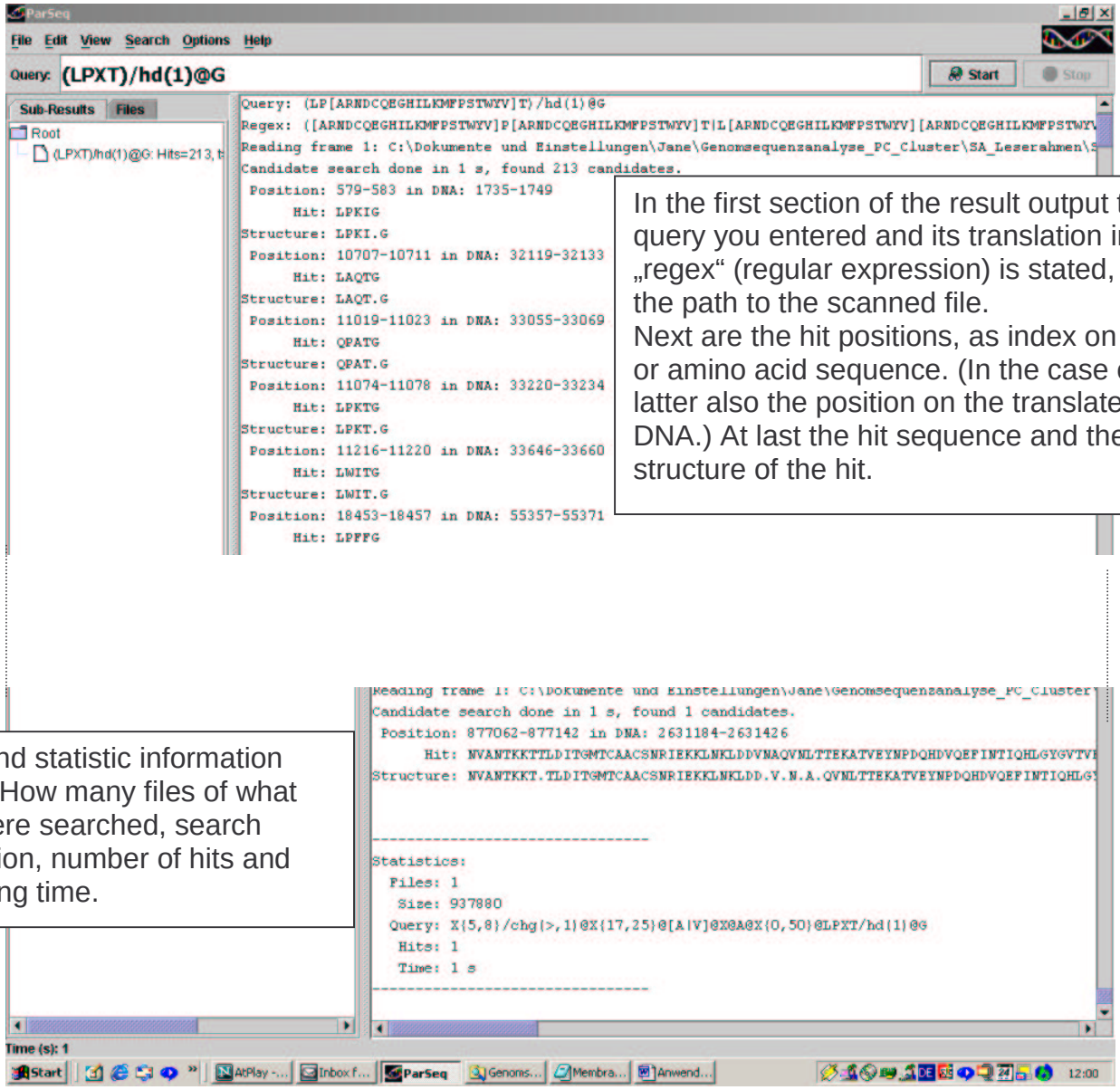
You can select a text file from the file directory or reaccess a saved selection or save the current selection. Furthermore, the program asks whether the file is a DNA- or a amino acid sequence and also wants to know the reading frame in the case of the latter.

Query



Type in the regular expression that you want to search for. The search is either run on the selected file (root) or on the intermediate result you specify by clicking on it.

Result output



In the first section of the result output the query you entered and its translation into the „regex“ (regular expression) is stated, as is the path to the scanned file.
Next are the hit positions, as index on DNA or amino acid sequence. (In the case of the latter also the position on the translated DNA.) At last the hit sequence and the structure of the hit.

At the end statistic information follows: How many files of what sizes were searched, search expression, number of hits and computing time.

6 Appendix

List of side conditions

Hamming-Distance: symbol: **hd**
Arguments: number of errors

Example:
LPXT/hd(1)@
You are searching for the amino acid pattern LPXT where a Hamming distance of 1 is allowed.
(Hamming distance of 1: One difference at a position is allowed, like XPXT or LXXT or LPXX or LPXT

Edit distance: symbol: ed
Argument: number of errors

Example:
LPXT/ed(1)@
You are searching for the amino acid pattern LPXT where an edit distance of 1 is allowed. (Edit distance of 1: One difference at a position is allowed (just as it is with the Hamming distance) or a deletion or an insertion e.g.: _PXT, LXPXT ...)

Biochemical attributes: symbols: see tables in the appendix.
Arguments: (bracket list, divided by commas):
window size (between 1 and 25)
logical operator (<, > or =)

Note: Insert a slash after the part of the pattern which is supposed to have the biochemical attribute and after that the corresponding symbol and the list of arguments.

The range of values conforms with the table that is to be used. The values of the individual AA can be looked up in the internet for all implemented scaling tables at the following address:http://us.expasy.org/cgi-bin/protscale.pl

Example:
X{5,8}/chg(>,2)@X{19,23}/hdp_kd(5, >, -1)@
You are searching for a section of 5-8 arbitrary AA with a positive charging greater than 2 (summation) and an attached section of 19 to 23 arbitrary AA which have an average hydrophobicity score greater than -1 according to Kyte/Doolittle within a window size of 5.

Used scaling lists and symbols

Criteria	Author	Article	Symbol
Hydropathicity	Kyte J., Doolittle R.F	J. Mol. Biol. 157:105-132(1982)	**hdp_kd**
Mokecular Weight		Most textbooks	**mcw_**
Bulkiness	Zimmerman J.M., Eliezer N., Simha R.	J. Theor. Biol. 21:170-201(1968)	**blk_ze**
Polarity	Grantham R	Science 185 :862-864(1974)	**pol_g**
Recognition factors	Fraga S.	Can. J. Chem. 60:2606-2610(1982)	**ref_f**
Optimized matching hydrophobicity (OMH)	Sweet R.M., Eisenberg D	J. Mol. Biol. 171:479-488(1983)	**omh_se**
Hydrophobicity (delta G1/2 cal)	Abraham D.J., Leo A.J	Proteins: Structure, Function and Genetics 2:130-152(1987)	**hdp_al**
Hydrophobicity (free energy of transfer to surface in kcal/mole)	Bull H.B., Breese K	Arch. Biochem. Biophys. 161:665-670(1974)	**hdp_bb**
Hydrophobicity scale based on free energy of transfer (kcal/mole)	Guy H.R.	Biophys J. 47:61-70(1985)	**hdp_g**
Hydrophobicity scale (contact energy derived from 3D data)	Miyazawa S., Jernigen R.L	Macromolecules 18:534-552(1985)	**hdp_mj**
Hydrophobicity scale (pi-r)	Roseman M.A	J. Mol. Biol. 200:513-522(1988)	**hdp_r**
Antigenicity value X 10	Welling G.W., Weijer W.J., Van der Zee R., Welling-Wester S	FEBS Lett. 188:215-218(1985)	**agv_wv**

Criteria	Author	Article	Symbol
Hydrophilicity scale derived from HPLC peptide retention times	Parker J.M.R., Guo D., Hodges R.S	Biochemistry 25:5425-5431(1986)	**hdp_pg**
Hydrophobicity indices at ph 7.5 determined by HPLC	Cowan R., Whittaker R.G	Peptide Research 3:75-80(1990)	**hdp_cw**
Retention coefficient in HFBA	Browne C.A., Bennett H.P.J., Solomon S	Anal. Biochem. 124:201-208(1982)	**rec_bb**
Retention coefficient in HPLC, pH 2.1.	Meek J.L	Proc. Natl. Acad. Sci. USA 77:1632-1636(1980)	**rec_m**
Molar fraction (%) of 2001 buried residues	Janin J.		**mfr_j**
Proportion of residues 95% buried (in 12 proteins)	Chothia C.	J. Mol. Biol. 105:1-14(1976).	**prb_c**
Atomic weight ratio of hetero elements in end group to C in side chain	Grantham R.	Science 185 :862-864(1974)	**whe_g**
Average flexibility index	Bhaskaran R., Ponnuswamy P.K.	Int. J. Pept. Protein. Res. 32 :242-255(1988)	**afi_bp**
Conformational parameter for beta-sheet	Chou P.Y., Fasman G.D	Adv. Enzym. 47:45-148(1978)	**pbs_cf**

Criteria	Author	Article	Symbol
Charge		http://speedy.embl-heidelberg.de/aas/	**chg_**
Conformational parameter for alpha helix	Deleage G., Roux B.	Protein Engineering 1:289-294(1987)	**pah_dr**
Conformational parameter for beta-turn	Deleage G., Roux B.	Protein Engineering 1:289-294(1987)	**pbt_dr**
Normalized frequency for alpha helix	Levitt M	Biochemistry 17:4277-4285(1978)	**fah_l**
Normalized frequency for beta-turn	Levitt M.	Biochemistry 17:4277-4285(1978)	**fbt_l**
Conformational preference for antiparallel beta strand	Lifson S., Sander C.	Nature 282:109-111(1979).	**abz_ls**
Overall amino acid composition (%).	McCaldon P., Argos P.	Proteins: Structure, Function and Genetics 4:99-122(1988)	**aac_ca**
Relative mutability of amino acids (Ala=100)	Dayhoff M.O., Schwartz R.M., Orcutt B.C	"Atlas of Protein Sequence and Structure", Vol.5, Suppl.3 (1978)	**maa_ds**
Polarity	Zimmerman J.M., Eliezer N., Simha R.	J. Theor. Biol. 21:170-201(1968)	**pol_ze**
Refractivity	Jones. D.D.	J. Theor. Biol. 50:167-184(1975)	**ref_j**
Conformational parameter for alpha helix (computed from 29 proteins).	Chou P.Y., Fasman G.D.	Adv. Enzym. 47:45-148(1978)	**pah_cf**
Conformational parameter for beta-turn(computed from 29 proteins).	Chou P.Y., Fasman G.D.	Adv. Enzym. 47:45-148(1978)	**pbt_cf**

Criteria	Author	Article	Symbol
Conformational parameter for beta-sheet.	Deleage G., Roux B.	Protein Engineering 1:289-294(1987)	**pbs_dr**
Conformational parameter for coil	Deleage G., Roux B.	Protein Engineering 1:289-294(1987).	**pco_dr**
Normalized frequency for beta-sheet	Levitt M	Biochemistry 17:4277-4285(1978)	**fbs_l**
Conformational preference for total beta strand (antiparallel+parallel)	Lifson S., Sander C.	Nature 282:109-111(1979)	**pbz_ls**
Membrane buried helix parameter	Rao M.J.K., Argos P	Biochim. Biophys. Acta 869:197-214(1986)	**mbh_ra**

Amino acid codes

Name	Symbol (1-letter code)	3-letter code
Glycin	G	Gly
Alanin	A	Ala
Valin	V	Val
Leucin	L	Leu
Isoleucin	I	Ile
Cystein	C	Cys
Methionin	M	Met
Phenylalanin	F	Phe
Tyrosin	Y	Tyr
Tryptophan	W	Trp
Prolin	P	Pro
Serin	S	Ser
Threonin	T	Thr
Asparagin	N	Asn
Glutamin	Q	Gln
Asparaginsäure	D	Asp
Glutaminsäure	E	Glu
Histidin	H	His
Lysin	K	Lys
Arginin	R	Arg

Abbreviations used in regular expressions

Combinations of AA

X=[ARNDCQEGHILKMFPSTWYV]

Combinations of nucleic acids

X=[AGTC]
N=[AGCT]
R=[GA]
Y=[TC]
K=[GT]
M=[AC]
S=[GC]
W=[AT]
H=[ACT]
B=[GTC]
V=[GCA]
D=[GAT]

Planned enhancements/upgrades

Boolean operators:
It shall be possible to connect parts of the regular expressions by Boolean operators. AND, NOT and OR will be implemented.
It should be possible to set up more conditions for a part of an expression and to define which characters or character sequences shouldn't occur in a part of an expression.

Errors:
Errors shall not only be allowed for parts of expressions but also for structured patterns,
e.g. if you search for PATTERN1-variable distance-PATTERN2 , one error shall be allowed, either in Pattern1 or Pattern2.
Moreover, it shall be possible to allow errors within a sequence with specific biological attributes. For instance a difference in the demanded limit value in a defined number of window positions.

Statistical measurements:
So far only the average values are used for the calculation of the limits within a window position. In order that single values don't get too much severity it should be possible to use other statistical measurements (standard deviation ...) as well.

Translation:
The program shall be able to translate a DNA sequence into the six possible amino acid sequences (six reading frames).

Implementaion of ORF limits:
Additionally a list of ORF limits shall be added to the program. You will then also be able to specify how far the pattern you are searching for may be away from the beginning and end of the sequence.
Biological background: Some essential parts of proteins, like signal sequences, are in defined closeness to the C- or N-terminating end. Regulatoric elements on the DNA have defined distances to the ORF limits. This information ought to be included in the search.

Complementary DNA:
Furthermore, an automatic search of patterns also on the complementary DNA shall be integrated and the option to put hits on the coding DNA into spacial reference to those on the complementary DNA.
Biological background: Binding sites for proteins on the DNA can be characterized by the fact that several sequence patterns on the coding DNA and on the complementary DNA are in direct proximity to each other.

Scan directories:
It shall be made possible to scan many DNA sequences successively. For instance, the input could be the address of a folder, all files in it shall then be scanned for the specified regular expression.
Biological background: It shall be possible to scan directories of eucaryal genes too.

Assorted hit list (ranking):
Emphasis of the hits found by „ degree to which the conditions are fulfilled“ or „Do allowed errors occur or does the pattern occur in the ideal shape ?“
The hits found shall be sorted according to that emphasis.

Integration of the ParSeq hits into an annotated presentation of the examined genome:
A viewer for the presentation of GenBank ®[1]-entries is written. It shall be made possible to interactively integrade ParSeq-hits into this presentation. Thus it will be possible to query annotation data and sequences of found ORF's with a mouse click.

Use of additional measurements for the examination of biochemical properties:
Use of „weighted matrices“ instead of scaling tables. Verification by external programs.

Involvement of external programs:
Transfer of hits to external programs which are able to check the hits for certain attributes, e.g. by using neural networks.
Biological background: The estimation of biochemical attributes merely on the basis of the amino acid sequence is sufficient in very few cases only. For many problems there are already specialized programs available which can provide better results due to a more complex approach. However, these programs are often unable to scan long sequences. Nevertheless, the hit sequences can be edited with them.

Speed-up of calculation time:
Parallelization of algorithms and implementaion on the Kepler cluster.[2]

[1] 1Or annotation data from other databases in the GenBank® format.
(GenBank®: Gene database of the NCBI – National Center for Biotechnology Information , USA
[2] 2http://kepler.sfb382-zdv.uni-tuebingen.de/kepler/start.shtml