EMBnet.news

MacOSX and Bioinformatics (part 2)
SARS report
Pftools version 2.3
Pattern discovery in regulatory sequences

Volume 9 Nr. 2
September 2003

Editorial

The SARS (Severe Acute Respiratory Syndrome) took us by surprise in March and the whole world was chocked and scared by this emerging threat. Thankfully the scientific community and the political authorities reacted fast enough to circumvent the disease.
The Bioinformatics community also was rapidly involved in the fight with the identification of the virus by BLASTing against the databases of known virus sequences and the creation of dedicated web sites. The Chinese EMBnet node was the first to set up such a web site followed by big members of the field like the NCBI, the EBI, and the Swiss-Prot database.
In this issue Prof. Luo gives us a short report on the topic.

Continuing the MacOSX and Bioinformatics section, we provide our readers with a full explanation on how to install a local Blast server, accessible both from the command-line and the default web server found in every MacOSX. This section also explains the tips and tricks to obtain different interfaces to the EMBOSS package, and the next issue will give an overview of the performance gain from Altivec optimised programs versus other plateforms. Many thanks to Erik Bongcam and Anders Nister!

Finally I encourage all readers to have a deep look at two software packages RSAT (regulatory sequence analysis tool) from Dr. van Helden and the long awaited Pftools release 2.3 from Dr. Philipp Bucher and Volker Flegel.

The editorial board:
Erik Bongcam-Rudloff, Laurent Falquet
Pedro Fernandes, Gonçalo Guimaraes Pereira

Protein Spotlight

Protein Spotlight (ISSN 1424-4721) is a periodical electronic review from the SWISS-PROT group of the Swiss Institute of Bioinformatics (SIB). It is published on a monthly basis and consists of articles focused on particular proteins of interest. Each issue is available, free of charge, in HTML or PDF format at http://www.expasy.org/spotlight

We provide the EMBnet community with a printed version of issue 36. Please let us know if you like this inclusion.

Contents

Editorial 2
Pattern discovery in regulatory sequences 3
Anti-SARS web site 10
Pftools version 2.3 12
MacOSX and Bioinformatics (Part II) 16
Protein spotlight 36 23
National nodes 25
Specialist nodes 27

Publisher:
EMBnet Administration Office
c/o Jack Leunissen
Dept. of Genome Informatics
Wageningen UR, Dreijenlaan 3
6703 HA Wageningen
The Netherlands
Email: Jack.Leunissen@wur.nl
Tel: +31 317 484074

Cover picture: Col de Fenêtre, near Grand-St-Bernard pass, Swiss Alps June 2002 [® Dmitri Kuznetov]

Pattern discovery in regulatory sequences : how to interpret the results ?

Jacques van Helden
Service de Conformation des Macromolécules Biologiques et de Bioinformatique.
Université Libre de Bruxelles, Campus Plaine, CP 263.
Bld du Triomphe.
B-1050 Bruxelles, Belgium.
e-mail: Jacques.van.Helden@ulb.ac.be

Abstract
In this paper, we illustrate the utilisation of two pattern discovery programs from the Regulatory Sequence Analysis Tools (`http://rsat.ulb.ac.be/rsat/`), for the prediction of regulatory elements in a set of co-expressed genes. The motifs discovered are then compared to known binding site stored in the SCPD database (`http://cgsigma.cshl.org/jian`).

Introduction
With the advent of microarray technology, biologists are now able to detect groups of genes characterized by a common response to some environmental (culture medium, drug, ...), or genetical (gene mutation) condition. The question is then to detect the cis-acting elements which mediate this co-expression. Several web services allow to predict regulatory motifs from sets of co-expressed genes:
`http://rsat.ulb.ac.be/rsat/`;
`http://www.lsi.upc.es/~alggen`;
`http://cbcsrv.watson.ibm.com/Tspd.html`;
`http://bighost.area.ba.cnr.it/BIG/PatSearch/`;
`http://bio.cs.washington.edu/software.html`;
`http://bayesweb.wadsworth.org/gibbs/gibbs.html`;
`http://bussemaker.bio.columbia.edu/reduce/`;
`http://www.esat.kuleuven.ac.be/inclusive`;
We will illustrate here some functionalities of the regulatory Sequence Analysis Tools (`http://rsat.ulb.ac.be/rsat/`) on the basis of a concrete example. Starting form a set of co-regulated genes, we will apply several pattern discovery methods, and check whether the programs are able to return relevant motifs.

Selection of a cluster of co-expressed genes
Expression data about genes responding to various carbon sources was imported from `http://www-genome.stanford.edu/yeast_stress` (Gasch et al., 2000). A chip-per-chip normalization was performed (median-centring, quartile-based estimation of dispersion), and genes were filtered with a lower Z-score threshold of 3.6 (corresponding to an E-value of 1). The remaining genes were clustered with Michael Eisen's Cluster software (Eisen et al., 1998) (`http://rana.lbl.gov/`), using average-linkage hierarchical clustering. The resulting tree was displayed with Treeview (part of Eisen's package), and a few groups of co-expressed genes were selected manually. Among them, we selected a group of 58 genes which are repressed by ethanol, and activated by all other carbon sources (Figure 1). This cluster will be used as study case to introduce the concepts of pattern discovery.

Sequence retrieval
The first access to Regulatory Sequence Analysis Tools (RSAT) is usually through the sequence retrieval program. More than 130 organisms are currently supported, and a default upstream size has been defined for each of them, on the basis of prior knowledge about regulation. We selected the organism Saccharomyces cerevisiae, entered the 58 gene names selected above, and obtained their upstream sequence over 800bp (the default size for the yeast). In addition

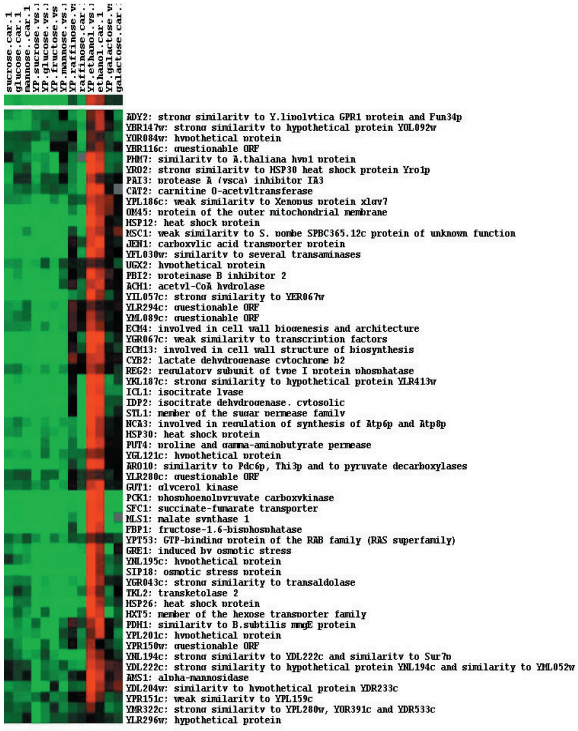


Figure 1: expression profiles of the 58 genes selected with Eisen's Cluster program.

to the default options, for the specific case of S.cerevisiae, we allowed overlap with upstream ORFs, because the annotations of S.cerevisiae contain around 1,000 false genes, located in intergenic regions, which leads to truncate many upstream sequences erroneously. Upstream ORFs should however be clipped for other organisms, especially for bacteria, where many genes are located inside operons.

Pattern discovery

The sequence retrieval result page contains a list of buttons, which allow to send the sequence to several pattern discovery programs.

oligo-analysis

We will start by applying the simplest approach : oligo-analysis (van Helden et al., 1998). This program counts the number of occurrences of all oligonucleotides of a specified size (hexanucleotides by default), and selects those which are significantly over-represented in the submitted sequences, as compared to the random expectation (i.e. if genes had been selected at random in the genome). Hexanucleotides are regrouped by pairs of reverse complements, because most yeast regulatory elements are strand-insensitive. In the following text, we refer to a pair of reverse complementary hexanucleotides as a pattern. Among the 2080 considered patterns, no more than 13 are significantly over-represented (Figure 2). The most significant hexanucleotide is CCCTTA|TAAGGG, found in 45 occurrences. The column 'expected occurrences' indicates that a random selection of 58 yeast upstream sequences would contain on the average 16.86 occurrences of this pattern. The probability to observe at least 45 occurrences when expecting 16.86 is 1e-08. However, this probability can be misleading, since each analysis involves simultaneous tests of 2080 patterns. To correct for this, the program returns an E-value (expected value), which indicates how many patterns would be expected at random, with such a level of over-representation. The pattern CCCTTA|TAAGGG has an E-value of 2.1e-05, which means that such a result would be expected by chance once per 50,000 trials (sequence sets). This pattern is thus highly significant.

Among the 13 selected patterns, some are mutually overlapping, and can be aligned to form larger motifs, as shown at the bottom of the result page (Figure 2). For example, the hexanucleotides gCGGCT and CGGCTa form a heptanucleotide gCGGCTa.

dyad-analysis

A more exhaustive analysis is performed by the program dyad-analysis (van Helden et al., 2000). This is an extension of oligo-analysis, which detects over-represented dyads. A dyad is defined here as a pair of short oligonucleotides (e.g. trinucleotides) separated by a spacing of fixed width but variable content. By default, all spacing values between 0 and 20 are considered. Among the 43,680 possible dyads, 35 are considered as significantly over-represented in the upstream sequences of the 58 ethanol-repressed genes (Figure 3). Several of these dyads are mutually overlapping, and can be assembled to form 7 larger patterns.

The most significant dyad, CCCN{0}TTA|TAAN{0}GGG, is actually identical to the first hexanucleotide detected by oligo-analysis. This is not surprising: dyad-analysis somehow encompasses oligo-analysis, since a hexanucleotide is nothing else than a pair of trinucleotides with a spacing of 0. On the contrary, the second dyad, CCGN{5}CCG|CGGN{5}CGG, could not be detected by oligo-analysis, due to the 5 nucleotide spacing between the two conserved parts. This dyad is found in 21 occurrences, whereas 4.51 would be expected at random. The E-value of 7.1e-04 indicates a high significance level.

Displaying pattern matches

The patterns detected by oligo-analysis or dyad-analysis can be searched in the input sequences, and their positions displayed on a feature-map (Figure 4). Each pattern is represented as a coloured box, with a thickness proportional to its significance.

Matching discovered patterns against SCPD

One would of course like to know which factors are likely to bind the discovered patterns. A very valuable resource for the yeast community is the Saccharomyces Cerevisiae Promoter Database (SCPD; `http://cgsigma.cshl.org/jian`).

The most significant motif, CCCTTA, matches two entries in SCPD : an UASCAR element found upstream the gene CAR2, and an element called 'heat _ shock _ not _ HSE' in DDR2. These two factors are documented by a little number of entries in the database, and it is hard to define a consistent consensus for them. The matching of the discovered pattern with these two elements might thus reflect the binding of one of them, but could also be spurious.

A more interesting result is obtained with the motif GGAN{5}GGA|TCCN{5}TCC, which matches 6

RSA-tools - oligo-analysis result

Information

One or several button(s) will appear at the bottom of this page, allowing you to send the result as input for a subsequent query.

Result

```
; oligo-analysis  -sort -format fasta -tho 1 -thosig 0 -return occ,rank,proba -2str -noov -v -seqtype dna -l 6 -bg upstream
-org Saccharomyces_cerevisiae -pseudo 0.05
; Citation: van Helden et al. (1998). J Mol Biol 281(5), 827-42.
; Detection of over-represented words
; Oligomer length              6
; Input format                 fasta
; Discard overlapping matches
; Counted on both strands
;              grouped by pairs of reverse complements
; Background model             upstream
; Organism                     Saccharomyces_cerevisiae
; Method                       Frequency file
; Expected frequency file      data/genomes/Saccharomyces_cerevisiae/oligo-frequencies/6nt_upstream_Saccharomyces_
cerevisiae-1str.freq
; Pseudo weight                0.05
; Pseudo frequency             2.40384615384615e-05
; Sequence type                DNA
; Nb of sequences              58
; Sum of sequence lengths      45097
; discarded occurrences        0               (contain other letters than ACGT)
; nb possible positions        44807
; total oligo occurrences      44807
; total overlapping occurrences728
; total non overlapping occ    44079
; alphabet size                4
; nb possible oligomers        2080
; threshold on occurrences     1
; threshold on occ sig         0
; threshold on occ proba       0.000480769230769231
; Sequences:
;               ADY2_0_800     800
(...)
;               YLR296w_653_147 147
;
; column headers
;               1              seq             oligomer sequence
;               2              identifier      oligomer identifier
;               3              expected_freq   expected relative frequency
;               4              occ             observed occurrences
;               5              exp_occ         expected occurrences
;               6              occ_P           occurrence probability (binomial)
;               7              occ_E           E-value for occurrences (binomial)
;               8              occ_sig         occurrence significance (binomial)
;               9              ovl_occ         number of overlapping occurrences (discarded from the count)
;               10             rank            rank
seq     identifier          expected_freq       occ   exp_occ    occ_P      occ_E      occ_sig    ovl_occ    rank
ccctta  ccctta|taaggg       0.0003781427313      45    16.86     1e-08      2.1e-05    4.68       0          1
ggggta  ggggta|tacccc       0.0002019404467      27    9.02      1e-06      2.1e-03    2.68       0          2
ccgaaa  ccgaaa|tttcgg       0.0004096816183      42    18.27     1.4e-06    2.9e-03    2.54       0          3
ccgcgc  ccgcgc|gcgcgg       0.0001371629722      21    6.13      2e-06      4.2e-03    2.38       2          4
aggggg  aggggg|cccccc       0.0001821106075      24    8.14      4.9e-06    1.0e-02    1.99       0          5
agccgc  agccgc|gcggct       0.0002587859854      28    11.56     3e-05      6.2e-02    1.21       0          6
cccgcg  cccgcg|cgcggg       0.0001163888551      17    5.21      3.2e-05    6.8e-02    1.17       0          7
acaagg  acaagg|ccttgt       0.0005099639475      43    22.74     9.8e-05    2.0e-01    0.69       0          8
cggcta  cggcta|tagccg       0.0002219591414      24    9.92      0.00011    2.2e-01    0.65       0          9
aaggga  aaggga|tccctt       0.0006546273451      51    29.17     0.00016    3.3e-01    0.49       0          10
gatccc  gatccc|gggatc       0.0001889094095      21    8.44      0.00019    4.0e-01    0.40       1          11
agggga  agggga|tcccct       0.0003209194813      30    14.33     0.00020    4.2e-01    0.38       0          12
aagggg  aagggg|cccctt       0.0003630342824      32    16.21     0.00034    7.1e-01    0.15       0          13
catccc  catccc|gggatg       0.0002567085738      25    11.47     0.00036    7.6e-01    0.12       0          14
cgggcc  cgggcc|ggcccg       0.0001305530259      16    5.84      0.00038    7.9e-01    0.10       2          15
aacaag  aacaag|cttgtt       0.0009889017758      68    43.97     0.00046    9.7e-01    0.02       0          16

; Job started 12/07/03 22:50:37 CEST
; Job done    12/07/03 22:51:06 CEST
```

Figure 2: pattern discovery result of oligo-analysis

sites in SCPD, 4 of which are bound by the Carbon Source Response Element (CSRE). Another discovered motif, GGGGTA|TACCCC matches two binding sites of Mig1p, the glucose repressor.

Checking the rate of false positive

An important advantage of the methods presented above is that they return a very low rate of false positive. Indeed, over-representation is assessed by applying a statistical test on each considered pattern independently. Thus, in principle, when a sequence set does not contain any over-represented pattern, oligo-analysis and dyad-analysis are able to return a negative answer. In theory, a pattern with an E-value <= 1 is expected once per trial (random sequence set),

Pattern assembly

```
; pattern-assembly  -v 1 -subst 1 -2str -i public_html/tmp/
oligo-analysis.2003_07_12.225035.res
; Input file     public_html/tmp/oligo-analysis.2003_07_
12.225035.res
; score column  8
; two strand assembly
; max flanking bases            1
; max substitutions             1
; number of patterns            16
;

;cluster # 1    seed: ccctta    4 words         length
;alignt         rev_cpl         score
tccctt.         .aaggga         0.49
tcccct.         .agggga         0.38
cccctt.         .aagggg         0.15
.ccctta         taaggg.         4.68
tccctta         taaggga         4.68            best consensus

;cluster # 2    seed: ccgcgc    2 words         length 7
;alignt         rev_cpl         score
cccgcg.         .cgcggg         1.17
.ccgcgc         gcgcgg.         2.38
cccgcgc         gcgcggg         2.38            best consensus

;cluster # 3    seed: aggggg    3 words         length 7
;alignt         rev_cpl         score
agggggg         ccccct          1.99
agggga          tcccct          0.38
aagggg          cccctt          0.15
agggggg         ccccct          1.99            best consensus

;cluster # 4    seed: agccgc    2 words         length 6
;alignt         rev_cpl         score
tagccg.         .cggcta         0.65
.agccgc         gcggct.         1.21
tagccgc         gcggcta         1.21            best consensus

;cluster # 5    seed: acaagg    2 words         length 7
;alignt         rev_cpl         score
aacaag.         .cttgtt         0.02
.acaagg         ccttgt.         0.69
aacaagg         ccttgtt         0.69            best consensus

;cluster # 6    seed: gatccc    2 words         length 7
;alignt         rev_cpl         score
gatccc          gggatc          0.40
catccc          gggatg          0.12
gatccc          gggatc          0.40            best consensus

; Isolated patterns: 3
;alignt         rev_cpl         score
ggggta          tacccc          2.68            isol
ccgaaa          tttcgg          2.54            isol
cgggcc          ggcccg          0.10            isol

;Job started 12/07/03 22:51:07 CEST
;Job done    12/07/03 22:51:08 CEST
```

Figure 2: (end)

and a pattern with E-value <= 1e-3 every 1,000 trials.

This theoretical model can be challenged either with randomly generated sequences, or with a random selection of real upstream sequences from the organism of interest. Both facilities are provided on the RSAT web site. The program random-seq generates random sequences, and allows to specify different residue probabilities (e.g. to generate AT-rich sequences). More biological-like sequences can be generated with Markov chain models, which mimic not only the residue frequencies, but also the oligonucleotide composition of biological sequences. The program allows for example to generate a random sequences with hexanucleotide frequencies calibrated on the full set of yeast intergenic sequences. Randomly generated sequences are useful for testing the accuracy of a program, but thy rely on some predefined probabilistic model.

A more realistic test can be performed by submitting a random selection of genes from the chosen organism. The program random-genes allows to select a set of random genes, which can then be sent to sequence retrieval and pattern discovery programs.

Discussion

The two approaches presented here are based on simple statistics (binomial test) on simple motifs (non-degenerated hexanucleotides, or spaced pairs of trinucleotides). Despite their simplicity, these approaches present several advantages over other pattern discovery tools, namely exhaustivity and low rate of false positive.

The programs oligo-analysis and dyad-analysis are exhaustive in the sense that, within the class of considered patterns, all possibilities are tested, and each significant result is returned. A single analysis is thus able to return binding sites for several transcription factors. In our test example, we identified putative binding sites for the carbon source response element, and for Mig1p, both of which are consistent with the conditions of expression (ethanol-repressed genes).

The low rate of false positive comes from the fact that the detection methods rely on a statistical test, which allows to select a threshold of significance. This is particularly important for the interpretation of gene expression data, since clustering methods are generally tricky, and the resulting clusters may often be artifactual. A program which would systematically return the 'best' motifs, without assessing whether they are or not significant, would thus predict one or several patterns in each of these artifactual clusters, whereas oligo-analysis and dyad-analysis will generally return no prediction.

Finally, it is important to realize that these tools (as most other pattern discovery tools) have until now been tested and used for microbial organisms. Their extension to higher organisms is far from trivial. Our preliminary tests indicate that many more false positives are returned for human than for yeast or bacterial sequences. This probably comes from the heterogeneity of human chromosomes, the presence of repeated elements, and the larger extensions of the sequences to analyse. we are currently developing more elaborate statistical models in order to address the complexity of transcriptional regulation in higher organisms.

RSA-tools - dyad-analysis result

Information

One or several button(s) will appear at the bottom of this page, allowing you to send the result as input for a subsequent query.

Result

```
; dyad-analysis  -v 1 -sort -return proba,rank -timeout 3600 -format fasta -type any -2str -noov -org Saccharomyces_
cerevisiae -thosig 0 -l 3 -spacing 0-20 -bg upstream
; Citation: van Helden et al. (2000). Nucleic Acids Res. 28(8):1808-18.
; Sequence type                 DNA
; Nb of sequences               58
; Sum of sequence lengths       45097
; default return values         proba,occ
; return values                 proba,occ,exp_freq,exp_occ,rank
; Monad parameters
;               monad size                      3
;               monad positions         89962
;                   valid                       89962
;                   discarded                   0 (contain other letters than ACGT)
;               distinct monads         64
; Dyad parameters
;               dyad type               any dyad
;               minimal spacing         0
;               maximal spacing         20
;               dyad positions          89614
;                   valid                       89614
;                   discarded                   0               (contain other letters than ACGT)
;               distinct dyads          43680
; Threshold on sig              0
; Estimation of expected dyad frequencies
;               Background model
;                   organism                    Saccharomyces cerevisiae
;                   sequence type               upstream
;               exp. freq. file                 data/genomes/Saccharomyces_cerevisiae/oligo-frequencies/dyads_3nt_sp0-20_
upstream_Saccharomyces_cerevisiae-1str.freq
; Sequences:
;               ADY2_0_800      800
(...)
;               YLR296w_653_147 147
; column headers
;               1               dyad_sequence   dyad sequence
;               2               dyad_identifier                 dyad identifier
;               3               expected_freq
;               4               obs_occ         observed occurrences
;               5               exp_occ         expected occurrences
;               6               occ_P           occurrence probability (binomial)
;               7               occ_E           E-value for occurrences (binomial)
;               8               occ_sig         occurrence significance (binomial)
;               9               ovl_occ         number of overlapping occurrences
;               10              all_occ         number of non-overlapping + overlapping occurrences
;               11              rank            rank
;               12              ov_coef         overlap coefficient
;               13              remark          remark
```

dyad_sequence	dyad_identifier	expected_freq	obs_occ	exp_occ	occ_P	occ_E	occ_sig	ovl_occ	all_occ	rank	ov_coef	remark
cccn{0}tta	cccn{0}tta|taan{0}ggg	0.0001785182508	45	15.80	2.2e-09	9.5e-05	4.02	0	45	1	1.0039	
ccgn{5}ccg	ccgn{5}ccg|cggn{5}cgg	0.0000506136859	21	4.51	1.6e-08	7.1e-04	3.15	5	26	2	1.0195	dir_rep
ggan{5}gga	ggan{5}gga|tccn{5}tcc	0.0001508407873	38	13.37	3.7e-08	1.6e-03	2.79	1	39	3	1.0156	dir_rep
tccn{6}cca	tccn{6}cca|tggn{6}gga	0.0001574418937	39	13.96	3.8e-08	1.7e-03	2.78	1	40	4	1.0000	
gggn{0}gta	gggn{0}gta|tacn{0}ccc	0.0000892591254	27	7.94	1e-07	4.5e-03	2.35	0	27	5	1.0039	
ctcn{10}ccc	ctcn{10}ccc|gggn{10}gag	0.0000974431639	28	8.66	1.7e-07	7.4e-03	2.13	0	28	6	1.0010	
ccgn{0}cgc	ccgn{0}cgc|gcgn{0}cgg	0.0000592410231	21	5.28	2.1e-07	9.3e-03	2.03	2	23	7	1.0674	
ccgn{6}cgc	ccgn{6}cgc|gcgn{6}cgg	0.0000502772460	19	4.48	3.1e-07	1.3e-02	1.87	1	20	8	1.0674	
cccn{8}gtc	cccn{8}gtc|gacn{8}ggg	0.0000716916704	23	6.38	3.2e-07	1.4e-02	1.85	0	23	9	1.0010	
aggn{0}ggg	aggn{0}ggg|cccn{0}cct	0.0000813073102	24	7.24	7.8e-07	3.4e-02	1.47	0	24	10	1.0000	
cctn{4}ccg	cctn{4}ccg|cggn{4}agg	0.0000759244097	23	6.76	8.5e-07	3.7e-02	1.43	0	23	11	1.0039	
cgcn{14}ccg	cgcn{14}ccg|cggn{14}gcg	0.0000613152051	20	5.46	1.4e-06	6.2e-02	1.20	1	21	12	1.0078	
cggn{4}ccg	cggn{4}ccg|cggn{4}ccg	0.0000461540490	24	4.11	1.9e-06	8.4e-02	1.08	2	26	13	2.0078	inv_rep
ccgn{5}tgc	ccgn{5}tgc|gcan{5}cgg	0.0000858232066	24	7.64	1.9e-06	8.5e-02	1.07	0	24	14	1.0049	
ccgn{0}aaa	ccgn{0}aaa|tttn{0}cgg	0.0002077411716	42	18.40	2.2e-06	9.6e-02	1.02	0	42	15	1.0000	
cccn{1}gcg	cccn{1}gcg|cgcn{1}ggg	0.0000632965815	20	5.64	2.3e-06	1.0e-01	1.00	1	21	16	1.0664	
gggn{5}gga	gggn{5}gga|tccn{5}ccc	0.0000934252622	25	8.31	2.6e-06	1.1e-01	0.95	2	27	17	1.0000	
cccn{0}gcg	cccn{0}gcg|cgcn{0}ggg	0.0000475120957	17	4.24	2.6e-06	1.1e-01	0.95	0	17	18	1.0664	
ccgn{17}tgc	ccgn{17}tgc|gcan{17}cgg	0.0000771935983	22	6.88	3.8e-06	1.6e-01	0.78	0	22	19	1.0049	
cggn{6}ggc	cggn{6}ggc|gccn{6}ccg	0.0000602926336	19	5.37	4.2e-06	1.8e-01	0.74	2	21	20	1.0049	
cggn{7}gcc	cggn{7}gcc|ggcn{7}ccg	0.0000738068363	21	6.58	6.3e-06	2.8e-01	0.56	3	24	21	1.0635	
ccgn{4}ccc	ccgn{4}ccc|gggn{4}cgg	0.0000567435062	18	5.06	6.8e-06	3.0e-01	0.53	1	19	22	1.0049	
ccgn{3}gtc	ccgn{3}gtc|gacn{3}cgg	0.0000522817596	17	4.66	8.8e-06	3.8e-01	0.42	0	17	23	1.0010	
cgcn{4}ccg	cgcn{4}ccg|cggn{4}gcg	0.0000523478824	17	4.67	8.9e-06	3.9e-01	0.41	0	17	24	1.0078	
atan{9}ggg	atan{9}ggg|cccn{9}tat	0.0001628688074	34	14.46	1e-05	4.4e-01	0.36	0	34	25	1.0000	
cggn{6}gga	cggn{6}gga|tccn{6}ccg	0.0000829274097	22	7.39	1.1e-05	4.9e-01	0.31	1	23	26	1.0000	
atan{1}ggg	atan{1}ggg|cccn{1}tat	0.0001576443162	33	14.00	1.3e-05	5.6e-01	0.26	0	33	27	1.0000	
cccn{5}aag	cccn{5}aag|cttn{5}ggg	0.0001382373794	30	12.28	1.6e-05	7.0e-01	0.16	1	31	28	1.0000	
agcn{0}cgc	agcn{0}cgc|gcgn{0}gct	0.0001248434983	28	11.10	1.7e-05	7.4e-01	0.13	0	28	29	1.0039	
ccgn{4}tcc	ccgn{4}tcc|ggan{4}cgg	0.0000791212269	21	7.05	1.7e-05	7.6e-01	0.12	1	22	30	1.0049	
gcgn{5}gcc	gcgn{5}gcc|ggcn{5}cgc	0.0000670181217	19	5.97	1.8e-05	7.8e-01	0.11	2	21	31	1.0625	
atgn{18}ggg	atgn{18}ggg|cccn{18}cat	0.0001124816378	26	10.01	2e-05	8.8e-01	0.06	0	26	32	1.0000	
cccn{3}aag	cccn{3}aag|cttn{3}ggg	0.0001400831879	30	12.45	2e-05	8.8e-01	0.05	0	30	33	1.0000	
cgcn{12}gtc	cgcn{12}gtc|gacn{12}gcg	0.0000559105979	17	4.99	2e-05	8.9e-01	0.05	0	17	34	1.0010	
cccn{1}taa	cccn{1}taa|ttan{1}ggg	0.0001548576743	32	13.75	2.2e-05	9.5e-01	0.02	0	32	35	1.0000	

```
;Job started 12/07/03 23:06:56 CEST
;Job done    12/07/03 23:09:31 CEST
```

Pattern assembly

```
; pattern-assembly  -v 1 -subst 0 -2str -maxfl 1 -subst 0 -i public_html/tmp/dyad-analysis.2003_07_12.230654.res
; Input file     public_html/tmp/dyad-analysis.2003_07_12.230654.res
; score column  8
; two strand assembly
; max flanking bases            1
; max substitutions             0
; number of patterns            35
;

;cluster # 1                    seed: ccctta                    2 words             length
;alignt                         rev_cpl                         score
ccctta.                         .taaggg                         4.02
cccntat                         atanggg                         0.26
cccttat                         ataaggg                         4.02                best consensus

;cluster # 2                    seed: ccgnnnnnccg               9 words             length 7
;         alignt                        rev_cpl                 score
ggcnnnnnnnccg...                ...cggnnnnnngcc                 0.56
.gccnnnnnnccg...                ...cggnnnnnggc.                 0.74
..ccgnnnnnccg...                ...cggnnnnncgg..                3.15
..ccgnnnnnncgc..                ..gcgnnnnnncgg..                1.87
..ccgnnnnccc....                ....gggnnnncgg..                0.53
...cggnnnnccg...                ...cggnnnnccg...                1.08
........cccngcg.                .cgcnggg........                1.00
.........cccgcg.                .cgcggg.........                0.95
..........ccgcgc                gcgcgg..........                2.03
ggccggnnccccgcgc                gcgcggggnnccggcc                3.15                best consensus

;cluster # 3                    seed: ggannnnngga               4 words             length 16
;     alignt                         rev_cpl                   score
tggnnnnnngga                    tccnnnnnncca                    2.78
.ggannnnngga                    tccnnnnntcc.                    2.79
.ggannnncgg.                    .ccgnnnntcc.                    0.12
..gacnnncgg.                    .ccgnnngtc..                    0.42
tggacnnncgga                    tccgnnngtcca                    2.79                best consensus

;cluster # 4                    seed: gggnnnnngga               3 words             length 12
;     alignt                         rev_cpl                   score
tggnnnnnngga                    tccnnnnnncca                    2.78
.gggnnnnngga                    tccnnnnnccc.                    0.95
.gggnnnncgg.                    .ccgnnnnccc.                    0.53
tgggnnnncgga                    tccgnnnnccca                    2.78                best consensus

;cluster # 5                    seed: cgcnnnnccg                8 words             length 12
;        alignt                        rev_cpl                 score
ggcnnnnnnccg..                  ..cggnnnnnngcc                  0.56
.gccnnnnnnccg..                 ..cggnnnnnggc.                  0.74
..ccgnnnnnccg..                 ..cggnnnnncgg..                 3.15
..ccgnnnnnncgc.                 .gcgnnnnnncgg..                 1.87
..ccgnnnnccc...                 ...gggnnnncgg..                 0.53
...cgcnnnnccg..                 ..cggnnnngcg...                 0.41
........cccngcg                 cgcnggg........                 1.00
.........cccgcg                 cgcggg.........                 0.95
ggccgcnnccccgcg                 cgcggggnngcggcc                 3.15                best consensus

;cluster # 6                    seed: cggnnnnnngga              7 words             length 15
;      alignt                         rev_cpl                  score
gcgnnnnnncgg.                   .ccgnnnnnncgc                   1.87
.cggnnnnncgg.                   .ccgnnnnnccg.                   3.15
.cggnnnnccg..                   ..cggnnnnccg.                   1.08
.cggnnnnngga                    tccnnnnnnccg.                   0.31
..ggannnnngga                   tccnnnnntcc..                   2.79
..ggannnncgg.                   .ccgnnnntcc..                   0.12
...gacnnncgg.                   .ccgnnngtc...                   0.42
gcggacnnccgga                   tccggnngtccgc                   3.15                best consensus

;cluster # 7                    seed: cccntaa                   2 words             length 13
;alignt                         rev_cpl                         score
ccctta.                         .taaggg                         4.02
cccntaa                         ttanggg                         0.02
cccttaa                         ttaaggg                         4.02                best consensus

; Isolated patterns: 16
;alignt                         rev_cpl                         score
ggggta                          tacccc                          2.35                isol
ctcnnnnnnnnnnccc                gggnnnnnnnnnngag                2.13                isol
cccnnnnnnnngtc                  gacnnnnnnnnggg                  1.85                isol
aggggg                          cccccct                         1.47                isol
cctnnnnccg                      cggnnnnagg                      1.43                isol
cgcnnnnnnnnnnnnnnccg            cggnnnnnnnnnnnnnngcg            1.20                isol
ccgnnnnntgc                     gcannnnncgg                     1.07                isol
ccgaaa                          tttcgg                          1.02                isol
ccgnnnnnnnnnnnnnnnntgc          gcannnnnnnnnnnnnnnncgg          0.78                isol
atannnnnnnnnggg                 cccnnnnnnnnntat                 0.36                isol
cccnnnnnaag                     cttnnnnnggg                     0.16                isol
agccgc                          gcggct                          0.13                isol
gcgnnnnngcc                     ggcnnnnncgc                     0.11                isol
atgnnnnnnnnnnnnnnnnnggg         cccnnnnnnnnnnnnnnnnncat         0.06                isol
cgcnnnnnnnnnnnngtc              gacnnnnnnnnnnnngcg              0.05                isol
cccnnnaag                       cttnnnggg                       0.05                isol

;Job started 12/07/03 23:09:31 CEST
;Job done    12/07/03 23:09:44 CEST
```

Figure 3 (pages 7 & 8) Pattern discovery result of dyad-analysis

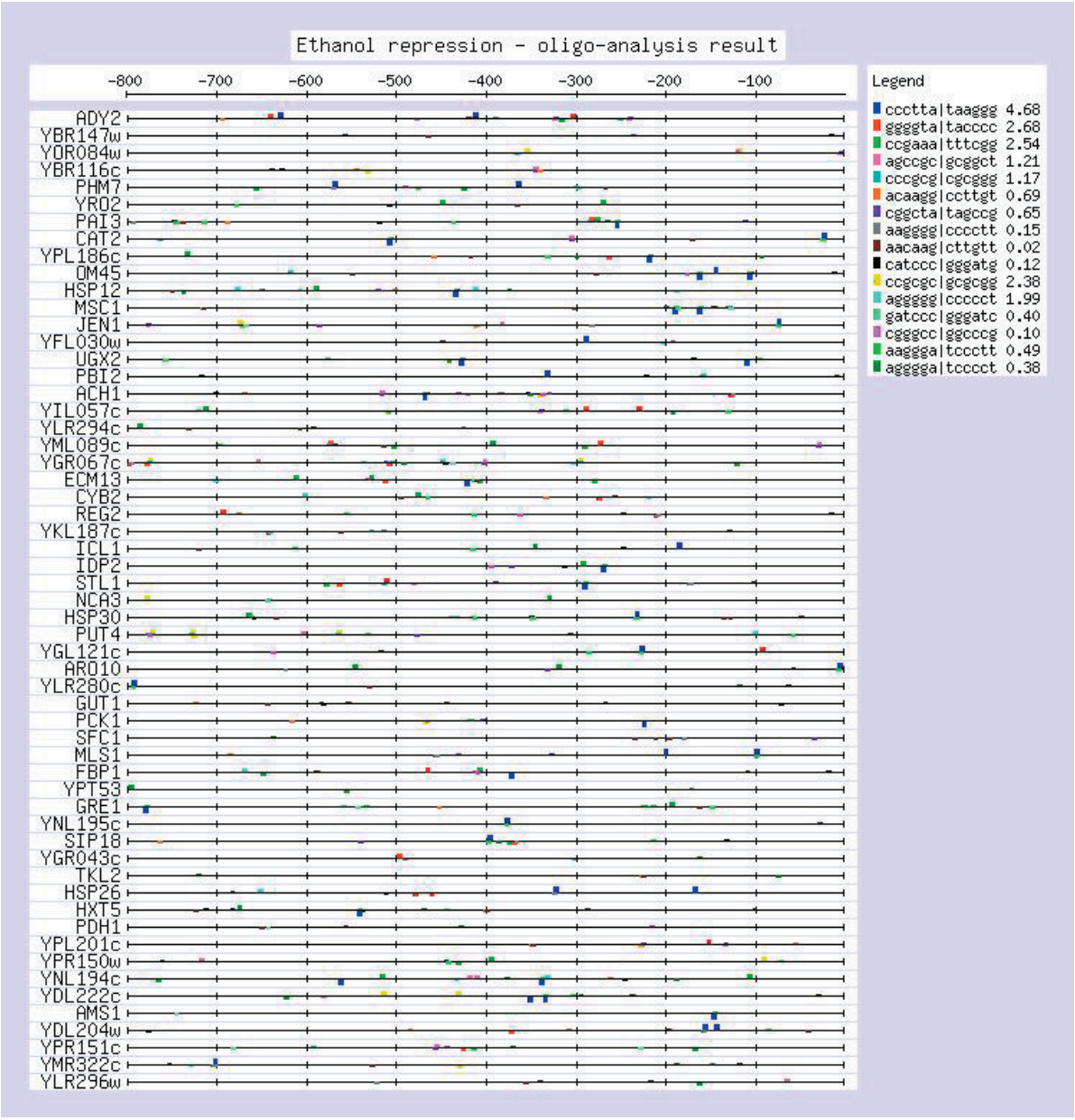


Figure 4: feature-map of the patterns discovered with oligo-analysis

References

1. Eisen, M. B., Spellman, P. T., Brown, P. O. & Botstein, D. (1998). Cluster analysis and display of genome-wide expression patterns. Proc Natl Acad Sci U S A 95(25), 14863-8.
2. Gasch, A. P., Spellman, P. T., Kao, C. M., Carmel-Harel, O., Eisen, M. B., Storz, G., Botstein, D. & Brown, P. O. (2000). Genomic expression programs in the response of yeast cells to environmental changes. Mol Biol Cell 11(12), 4241-57.
3. van Helden, J., André, B. & Collado-Vides, J. (1998). Extracting regulatory sites from the upstream region of yeast genes by computational analysis of oligonucleotide frequencies. J Mol Biol 281(5), 827-42.
4. van Helden, J., Rios, A. F. & Collado-Vides, J. (2000). Discovering regulatory elements in non-coding sequences by analysis of spaced dyads. Nucleic Acids Res 28(8), 1808-18.

Anti-SARS web site

Prof. Jingchu Luo
Chinese EMBnet node manager
Centre of Bioinformatics
Peking University
Beijing 100871
China
luojc@plum.lsc.pku.edu.cn

Introduction

The outbreak of the Severe Acute Respiratory Syndrome (SARS) starting from southern China early this year has a significant influence on public health as well as daily life. More than 100 new SARS cases were reported every day in late April from the nation's capital with a population of 12 million people. A nation wide anti-SARS battle started. The identification of SARS coronavirus (SARS-CoV) as the major causative factor of the SARS disease and the genomic sequences of the SARS virus made it possible for bioinformatics study. As the national node (1) of the European Molecular Biology Network (EMBnet), we have the responsibility to provide bioinformatics resource for the biological and medical researchers to assist their experimental investigation. An anti-SARS mailing list was created in the later April. Hundreds of SARS related news have been posted to more than 300 scientists working in the frontier. A dedicated anti-SARS bioinformatics web site was setup (2). A special SRS database server with sequence and structure information of SRAS-CoV and other viruses was installed, literature information and latest news were collected and made online, analysis results including multiple sequence alignment, structure prediction and phylogeny analysis in both English and Chinese were published. Various links to SARS related web sites such as the EBI 2CAN SARS page and the NCBI SARS genome page were collected. This web site was also linked in the EBI and NCBI web site. Collaboratively with colleagues at EBI and NCBI, we have been trying our best to make our contribution to the anti-SARS research during the past two months. In addition to the above web

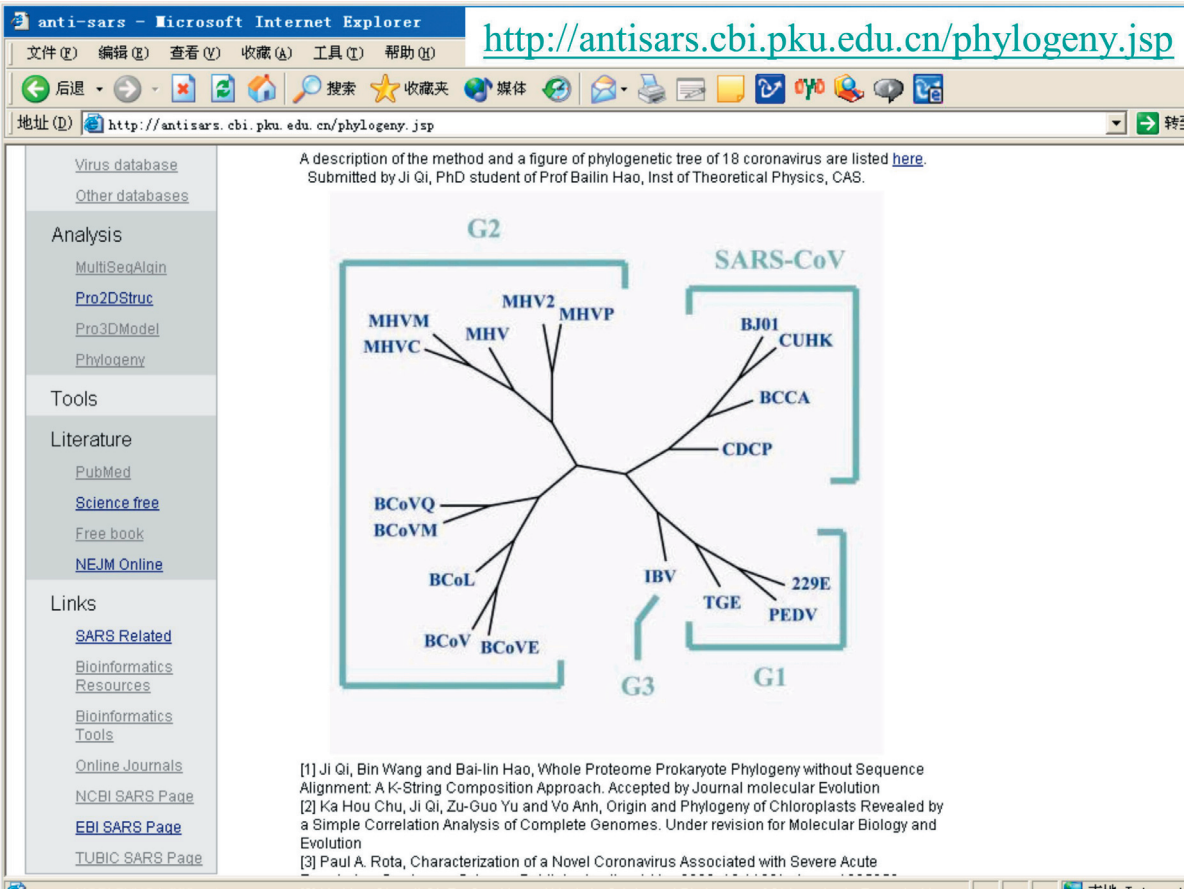


Figure 1. Example of the web server, Phylogeny diagram of the Corona viruses

Table of 42 mismatches among 12 complete SARS corona virus genome sequences

No	Pos	Freq	Sequence*												Codon	protein	AA
			CA1	BJ1	US1	HK1	HK2	HK3	SG1	SG2	SG3	SG4	SG5	TW1			
1	1476	1	a	a	a	a	a	a	a	a	a	a	G	a	AGA-AGG	P65-225	R-R
2	2601	1	t	t	t	t	C	t	t	t	t	t	t	t	GTT-GTC	P65-600	V-V
3	3165	1	a	a	a	a	a	a	a	a	a	a	a	G	TCA-TCG	Nsp1-149	S-S
4	7746	1	g	g	g	T	g	g	g	g	g	g	g	g	CCG-CCT	Nsp1-1676	P-P
5	7919	1	c	c	T	c	c	c	c	c	c	c	c	c	GCT-GTT	Nsp1-1734	A-V
6	7930	1	g	g	g	g	A	g	g	g	g	g	g	g	GAC-AAC	Nsp1-1738	D-N
7	8387	1	g	g	g	g	C	g	g	g	g	g	g	g	AGT-ACT	Nsp1-1890	S-T
8	8417	1	g	g	g	g	C	g	g	g	g	g	g	g	AGA-ACA	Nsp1-1900	R-T
9	8572	1	g	T	g	g	g	g	g	g	g	g	g	g	GTA-TTA	Nsp1-1952	V-L
10	9404	2	t	C	t	C	t	t	t	t	t	t	t	t	GTT-GCT	Nsp1-2229	V-A
11	9479	1	t	t	t	C	t	t	t	t	t	t	t	t	GTA-GCA	Nsp1-2254	V-A
12	9854	1	c	T	c	c	c	c	c	c	c	c	c	c	GCC-GTC	Nsp1-2379	A-V
13	10587	1	a	C	a	a	a	a	a	a	a	a	a	a	ACA-ACC	3cl-201	T-T
14	13494	1	g	g	g	g	A	g	g	g	g	g	g	g	GTT-AGT	RdRp-42	V-S
15	13495	1	t	t	t	t	G	t	t	t	t	t	t	t	GTT-AGT	RdRp-42	V-S
16	16622	1	c	c	T	c	c	c	c	c	c	c	c	c	GCC-GCT	Nsp10-152	A-A
17	17564	2	t	G	t	G	t	t	t	t	t	t	t	t	GAT-GAG	Nsp10-466	D-E
18	17846	2	c	c	c	T	c	T	c	c	c	c	c	c	CGC-CGT	Nsp10-560	R-R
19	18065	1	g	g	g	g	A	g	g	g	g	g	g	g	AAG-AAA	Nsp11-32	K-K
20	18282	1	c	c	c	c	c	c	A	c	c	c	c	c	CTA-ATA	Nsp11-105	L-I
21	18965	1	t	t	t	t	t	t	t	t	t	A	t	t	ATT-ATA	Nsp11-332	I-I
22	19064	2	a	a	G	G	a	a	a	a	a	a	a	a	GAA-GAG	Nsp11-365	E-E
23	19084	4	c	c	c	c	c	c	c	T	T	T	T	c	ACA-ATA	Nsp11-372	T-I
24	19838	1	a	G	a	a	a	a	a	a	a	a	a	a	GTA-GTG	Nsp12-96	V-V
25	21721	2	g	A	g	A	g	g	g	g	g	g	g	g	GGC-GAC	Spike-77	G-D
26	22222	2	t	C	t	C	t	t	t	t	t	t	t	t	ATT-ACT	Spike-244	I-T
27	23174	1	c	c	c	c	c	c	T	c	c	c	c	c	TCC-TCT	Spike-561	S-S
28	23220	1	G	t	t	t	t	t	t	t	t	t	t	t	TCT-GCT	Spike-577	S-A
29	23792	1	c	c	c	c	c	c	c	c	c	T	c	c	GTC-GTT	Spike-767	V-V
30	24872	1	t	t	C	t	t	t	t	t	t	t	t	t	CTT-CTC	Spike-1127	L-L
31	25298	1	A	g		g	g	g	g	g	g	g	g	g	GGA-AGA	Sars3a-11	G-R
32	25569	1	t	t	t	t	A	t	t	t	t	t	t	t	ATG-AAG	Sars3a-101	M-K
33	25673	1	a	C	a	a	a	a	a	a	a	a	a	a	AAG-CAG	Sars3a-136	K-Q
34	26050	1	a	C	a	a	a	a	a	a	a	a	a	a	CCA-CCC	Sars3b-261	P-P
35	26428	1	g	g	g	g	g	g	g	g	A	g	g	g	GAG-AAG	Membrane-11	E-K
36	26477	1	t	t	t	t	t	G	t	t	t	t	t	t	TTC-TGC	Membrane-27	F-C
37	26600	1	c	c	c	c	T	c	c	c	c	c	c	c	GCT-GTT	Membrane-68	A-V
38	26857	1	t	t	C	t	t	t	t	t	t	t	t	t	TCC-CCC	Membrane-154	S-P
39	27111	1	a	a	a	a	a	a	a	a	a	a	G	a	GAG-GGG	Sars6-13	E-G
40	27243	1	c	T	c	c	c	c	c	c	c	c	c	c	CCT-CTT	Sars6-57	P-L
41	27827	2	t	C	t	C	t	t	t	t	t	t	t	t	TGC-CGC	Sars8a-17	C-R
42	28696	1	g	g	g	g	g	T	g	g	g	g	g	g	GGT-TGT	Nucleocapsid-193	G-C
	Total	52	2	12	5	9	9	3	2	1	2	3	3	1			

*CA1= TOR2, BJ1=BJ01,US1=Urbani, HK1=CUHK-W1, HK2=HKU-39849, HK3=CUHKSu10, SG1=SIN2679, SG2=SIN2748, SG3=SIN2500, SG4=SIN2774, SG5=SIN2677, TW1-TW01. Colour code for amino acid change: green - same, yellow - similar, red - dissimilar. Colour code for number of changes: dark blue - 4, light blue - 2, light green - 1.

service, we have made bioinformatics analysis on the SARS-CoVs. Together with NCBI and EBI colleagues, we have published a paper «Initial analysis on the genome sequence of the SARS coronavirus» (3).

References

1. Chinese EMBnet node (http://www.cbi.pku.edu.cn)
2. Anti-SARS web site (http://antisars.cbi.pku.edu.cn)
3. Chen YJ, Gao G, Bao YM, Lopez R, Wu JM, Cai T, Ye ZQ, Gu XC, Luo JC (2003) Initial analysis on the genome sequence of the SARS coronavirus. Acta Genetica Sinica 30(6): 493~500.

Pftools version 2.3

Volker Flegel
PROSITE group
Swiss Institute of Bioinformatics
155, ch. des Boveresses
CH-1066 Epalinges,
Switzerland
Volker.Flegel@isb-sib.ch

Introduction

Pairwise sequence comparison has become a fundamental paradigm in biological sequence analysis. Many different computational tools have been developed in order to search for sequences in databases which display significant similarity between each other. Nevertheless, during the course of sequence evolution a great deal of variability has often been introduced in sequences belonging to the same functional or structural family. This renders pairwise sequence comparison inefficient, especially if only some functional residues (*i.e.* active site) are conserved. A simple way to tackle this problem is the use of *position specific scoring matrices* (PSSM). These matrices are derived from multiple sequence alignments and therefore include information about the variability or conservation of all residues at a given position of the alignment. This technique was further improved by the use of probabilistic models like *hidden Markov models* (HMM) and generalised profiles. These biological motif descriptors give a probability (or a score, for the generalised profiles) to each residue at a given position of the motif as well as a position specific probability for insertions and deletions in the alignment. Hence these motif descriptors allow the biologist to push his search further back on the evolutionary time scale, because they are able to find very distantly related members of a sequence family. In this short article we will describe a set of tools which allow the construction and use of generalised profiles as found in the PROSITE database.

Generalised profiles

These motif descriptors combine many aspects of PSSMs and certain types of HMMs. They can be used to search for promoter elements or other motifs on DNA sequences but they are more commonly applied to the identification of protein domains or families. The main repository for protein profiles is the PROSITE database maintained at the Swiss Institute of Bioinformatics.

Generalised profiles (hereafter simply called profiles) are composed of alternating match and insert positions (see figure 1). With each position is associated a set of scores. The match positions give a residue specific match extension score as well as a deletion extension score. When aligning a sequence to a profile, these scores are used when either a sequence residue is considered to match the given profile position or this particular position seems to be deleted in the sequence. The insert positions supply residue specific insert extension scores, as well as 16 values describing all the different transitions from one state to another (*begin, match, delete, insert* or *end*). The two begin and end states present at each insert position allow the profile to function in either a global alignment mode or several kinds of local alignment modes. The insert positions provide scores to residues inserted into the matching sequence as compared to the profile. They can also allow or penalise transitions between the states of the alignment (*i.e.* forbidding insertions at a specific position, by giving a negative score to match to insert transitions). Further information about generalised profiles can be found in Bucher 1994 and Bucher 1996.

For each alignment between a sequence and a profile, a score can be computed by summing all

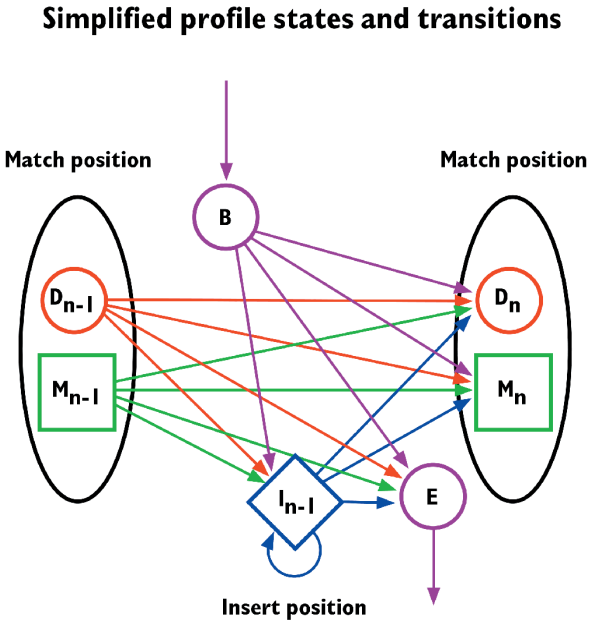


Figure 1. This diagram shows the basic building blocks of a PROSITE profile. The match positions are composed of a match (green) and a delete (red) state. An insert position is found between two successive match positions. The match and insert states provide scores for the complete residue alphabet, whereas only one score exists for the delete state. The transition scores are defined at the insert position. The begin (B) and end (E) states allow internal initiation or termination of the alignment.

the different state scores as well as the different state transitions. The goal being to find the best scoring alignment(s). A set of programs to build and search with generalised profiles is provided by the **pftools** package, which we will describe in the following paragraphs.

Pftools package description

The **pftools** package was originally developed by Philipp Bucher and is currently maintained at the Swiss Institute of Bioinformatics, Lausanne. These tools implement profiles in a machine-readable text file format as used by the PROSITE protein motif database and detailed at http://www.expasy.org/txt/profile.txt.
The package contains a set of programs to create, convert and search with profiles. It is distributed as FORTRAN sources which should readily compile on Unix like systems providing a FORTRAN compiler. Several precompiled binary packages will also be made available. The sources can be downloaded via anonymous ftp at: ftp://ftp.isrec.isb-sib.ch/pub/software/unix/pftools. A web page providing support and tutorials will be made available as soon as possible at the URL: http://www.isrec.isb-sib.ch/pftools.

The distribution contains 12 tools which are briefly described below.

2ft 2-frame interleaved translation of a DNA sequence into protein.
This tool translates both strands of a DNA sequence into protein by interleaving, for each strand, the translated residues of each reading frame. It is used to translate genomic or EST sequences to search against frame-search profiles generated by **ptof** (see below), thus allowing to easily identify coding regions or frame-shifts.

6ft 6-frame translation of a DNA sequence into protein.
As opposed to the **2ft** program, **6ft** does not generate interleaved protein sequences, but simply translates all 6 reading frames of a DNA sequence into protein.

gtop convert a profile in GCG format into PROSITE format.
Profiles created with the GCG program **ProfileMake** can be converted into the PROSITE profile format without data loss.

htop convert a HMMER ASCII-formatted HMM into an equivalent PROSITE profile.
This tool implements the method described by Bucher 1996 to convert a HMMER HMM motif descriptor into an equivalent PROSITE profile. Both HMMER1 and HMMER2 ASCII HMMs are supported. This conversion is useful to bridge the gap between the Pfam (relying on HMMs) and PROSITE motif databases.

pfmake generate a profile from a multiple sequence alignment.
This central program of the pftools package reads a weighted multiple sequence alignment in either MSF or MSA format, and computes a PROSITE profile according to the method described in Bucher 1994. Several parameters control the type of profile created (i.e. linear, circular, global-, local-alignment mode, etc.).

pfscale scale the parameters of a profile's normalisation function.
Given a sorted score distribution obtained by searching a sequence database with a profile, this program computes the parameters of an extreme-value distribution and rescales the profile's normalisation function accordingly.

pfscan scan a protein or DNA sequence against a profile library.
This is the first of two profile searching tools. It scans an input sequence for the occurrence of any profile stored in a given profile library. Matches are reported if the raw or normalised score exceeds a given cut-off level. The description of the alignment varies according to several command line parameters.

pfsearch search with a profile against a protein or DNA sequence library.
This second search tool finds the occurrences of an input profile in the sequences of a protein or DNA library. Similar to pfscan, matches are reported if their score exceeds a given cut-off value. The format of the output depends on several command line parameters.

pfw weight sequences of a multiple alignment.
This tool computes new weights for individual sequences in a multiple sequence alignment to compensate for the skew in the number of different family members in the original sequence alignment.

psa2msa convert a PSA file into a FASTA multiple sequence alignment file.
It reformats a PSA profile-sequence alignment as produced by **pfscan** or **pfsearch** into a Pearson/FASTA-formatted multiple sequence alignment (also called MSA). The output can then be reformatted through third party tools like **readseq** into other formats, or directly used by **pfmake** to create a PROSITE profile.

ptof convert a protein profile into a frame-search profile.
The frame-search profile created by **ptof** can be used to search for occurrences of a protein motif in DNA sequences (*i.e.* genomic or EST) previously translated with **2ft**.

ptoh convert a PROSITE profile into an approximately equivalent HMM.
This tool uses the reverse technique of **htop** to convert a PROSITE profile into an HMM compatible either with the SAM or HMMER1 package.

The concurrent use of these tools allows the easy creation and search for sequence motifs. They help the biologist to identify biological family members or discover functional elements in sequences. The conversion between different kinds of motif descriptors opens a whole range of possibilities to explore the structure of the biological sequence under investigation.
In the following paragraphs we will focus on the changes introduced between releases 2.2 and 2.3 of the **pftools** package.

Release 2.3 generalities

Some improvements, not necessarily visible to the user, affect the handling of input files. All programs included in the **pftools** package are now able to handle larger sequences or profiles compared to the previous release. Indeed, with the advent of genomic sequencing, the size of sequences submitted to these tools has increased during recent years.
An effort was made to catch and report to the user parse errors occurring on input files or problems that may arise during computation. Many of these errors would terminate the programs silently in release 2.2, leaving the user in doubt about the correct completion of the task. These error or warning messages are output on standard error and can be easily redirected or discarded.
The command line syntax has been rendered more Unix friendly. All options are now specified via a hyphen (**-**) followed by a letter and possibly a parameter value, as is common in the Unix world. The syntax used in release 2.2 is still supported but deprecated; it may not be so for future releases of the **pftools** package.
Programs having as output alignments, profiles or sequences now include an option to change the width of the output to adapt it to the users needs, *i.e.* for sequences the number of residues to display on one line can be set through the command line option **-W**.
Due to specifics of the formatted output in FORTRAN, numerical overflow could result in the display of a series of '*****' characters in place of the expected value. This was corrected wherever possible in release 2.3 of the **pftools**.
In addition we also corrected the known bugs of the different programs, which nevertheless did not lead to major changes in the algorithms used by the **pftools**.
We also introduced a controlled vocabulary, implemented as *keyword=value* pairs, to describe alignments while remaining compatible with the standard FASTA sequence header. This header syntax is called **xpsa** and will be detailed below (see also the man page included in the **pftools** distribution). The need for this kind of syntax arose when transferring information between and through different sequence analysis tools, most of them relying on sequences in FASTA format. Further developments around the **pftools** will rely more heavily on this kind of syntax. In the current release, **xpsa** is essentially used for output, but a tool like **pfmake** is able to read the weight of input sequences through the use of the '*weight=<value>*' keyword.
In the next paragraphs we describe some of the more commonly used tools of the package and the major changes between the release 2.3 and 2.2.

Creating a profile

Several tools are useful when generating profiles from multiple sequence alignments. Here we briefly describe some changes found in **psa2msa**, **pfw**, **pfmake** and **pfscale**.
psa2msa is used to convert alignments produced

```
>PEX7_HUMAN|O00628/63-323 motif=PS50294|WD_REPEATS_REGION
norm_score=38.968 raw_score=1376 match_type=region seq_end=-1
SFDWNDGLFDVTWSENnEHVLITCSGDGSLQLWDTAKAAGPLQvYKEHAQEVYSVDWSQT
rgEQLVVSGSWDQTVKLWDPTVGKSLCTFRGHESIIYSTIWSPHiPGCFASASGDQTLRI
WDVKAAGVRIVIPAHQAEILSCDWCKYnENLLVTGAVDCSLRGWDLRNVRQPVFeLLGHT
YAIRRVKFSPFhASVLASCSYDFTVRFWNFSKPDSLLEtVEHHTEFTCGLDFSLQsPTQV
ADCSWDETIKIYDPACLTIPA-
>PEX7_HUMAN|O00628/63-105 motif=PS50294|WD_REPEATS_REGION
norm_score=7.437 raw_score=180 repeat_nb=1 match_type=repeat
seq_end=-219
SFDWNDGLFDVTWSENnEHVLITCSGDGSLQLWDTAKAAGPLQ
[...]
>PEX7_HUMAN|O00628/282-323 motif=PS50294|WD_REPEATS_REGION
norm_score=4.353 raw_score=63 repeat_nb=6 match_type=repeat
seq_end=-1
VEHHTEFTCGLDFSLQsPTQVADCSWDETIKIYDPACLTIPA-
```

Figure 2. This example shows the alignment between a circular profile and a protein sequence. Elements of the xpsa header are highlighted in blue. The first alignment is the total match of the circular profile on the sequence (as identified by the 'match _ type=region' element). This alignment can be decomposed into 6 individual matches listed below the total match (only two have been included in this figure). They are characterised by the 'match _ type=repeat' and the 'repeat _ nb' elements.

by either **pfscan** or **pfsearch** (launched with option **-x**) into a Pearson/FASTA formatted multiple sequence alignment (MSA). It also allows to control the length of the resulting alignment by truncating insertions to a certain size. The behaviour of this option was changed in the new release: a value of '0' now removes all insert positions and the default value of '-1' imposes no limit on the length of insert regions.

The programs **pfw** and **pfmake** can now directly use these MSA files as input without requiring another conversion step to obtain an alignment in MSF format. The sequence weighting tool **pfw** will output the weighted multiple alignment in the same format as the input file. When the MSA format is used, the weight of each individual sequence is specified in the FASTA header using the '*weight=<value>*' keyword. This information can then be recovered by **pfmake** for profile generation.

When scaling the normalisation function of a profile by using **pfscale**, it is now possible to specify which normalisation mode is to be scaled (and implicitly all the cut-off level scores using this normalisation mode will be updated as well). Note that the current version of **pfscale** does not support input score lists in **xpsa** format.

Searching with profiles

The two search tools **pfsearch** and **pfscan** have very similar behaviours, the first searches with a profile against a sequence library whereas the second searches a sequence against a profile library. Therefore the new features have been similarly implemented in both programs.

Alignments are now printed correctly when option **-a** (report optimal alignment for all sequences regardless of the cut-off value) or **-u** (forces DISJOINT=UNIQUE) is combined with any of the **-s**, **-x** or **-y** output modifier options.

If a profile contains several normalisation modes it is possible to specify via the command line option **-M** which mode should be used for alignment score computation. This also overrides the profile's PRIORITY parameter which can be used to give a precedence for one normalisation mode over another.

The option **-k** enables the output of alignments with a **xpsa** header. As already mentioned, this is a more structured variation of the FASTA sequence header using *keyword=value* pairs to store information. It's syntax is detailed in the man page distributed with the pftools package. Briefly, the header must start with a '**>**' immediately followed by a sequence identifier. Alignment begin and end positions can be appended to this field using the '/' character. Then follows a list of *keyword=value* pairs which describe the raw score of the alignment, the normalised score, the motif identifier, etc. For examples refer to figure 2. This also allows to have a similar header structure when using **pfsearch** or **pfscan**, which is not the case when using the standard output format. Indeed **pfsearch** will report first the sequence identifier whereas **pfscan** uses the profile's identifier instead.

If a profile is circular, each match between a sequence and this profile can be composed

of repeats of individual matches. Release 2.3 of **pfsearch** and **pfscan**, can now report the individual matches for such circular profiles (see figure 2). Note that the scoring system of a circular profile was optimised to find total matches, therefore the normalised scores of the individual matches should be considered with caution.

Conclusion and future developments

The release 2.3 of the **pftools** package corrects many known bugs of the previous release. The tools should be less error prone due to an improved parsing of input files and buffer size tests. No new algorithm was developed for this release but many options have been added to the different programs in order to increase their flexibility and interoperability. This includes the creation of the **xpsa** header format, designed to allow the flow of information between different sequence analysis tools.

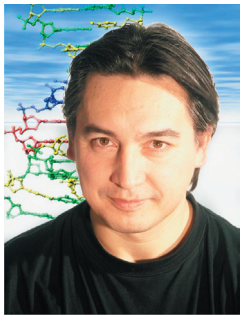
In order to allow easy implementation and testing of new algorithms, the main focus for future developments of the **pftools** will be to port the FORTRAN sources to C/C++. Emphasis will be set on developing a modular, library based design, thus enabling profile parsing and alignment functions to be easily used in other programs written in different languages. Several new sequence weighting, profile construction and scaling algorithms are currently tested and will eventually be included in future releases.

Further information can be obtained by mail at: pftools@isb-sib.ch

References

1. Bucher, P. and Bairoch, A. (1994) A generalized profile syntax for biomolecular sequence motifs and its function in automatic sequence interpretation. proc Int Conf Intell Syst Mol Biol, 2, 53-61.
2. Bucher, P., Karplus, K., Moeri, N. and Hofmann, K. (1996) A flexible motif search technique based on generalized profiles. Comput Chem, 20, 3-23.
3. Luthy, R., Xenarios, I. and Bucher, P. (1994) Improving the sensitivity of the sequence profile method. Protein Sci, 3, 139-46.

A guide on how to install Bioinformatics tools on MacOSX (Part II)



Erik Bongcam-Rudloff
Assistant Professor

Anders Nister
Project Student

Swedish University of Agricultural Sciences
The Linnaeus Centre for Bioinformatics

This is the second part in our tutorial series aimed at building a "Bioinformatics Workbench" on the MacOSX operating system. In this part we will learn how to install a web interface to the blast programs. You can also use the command line blast programs installed during our first part, but for the inexperienced Unix-users it is much easier to use a cut and paste system than using the terminal.

We will also learn how to install a graphical interface to the most popular bioinformatics package namely EMBOSS.

Installing your own databases to use with BLAST

To find the BLAST databases the blast binaries look for a file called ".ncbirc". The command line binaries look at the environment variable NCBI to be set to the directory where the .ncbirc file is located.

If you use our distribution of the ncbi-tools and ncbi-blast this file is located in the /usr/local/blast/ directory, and the NCBI environment variable should already be set for you (/usr/local/blast). Edit this file with a text editor to reflect where you store your databases.

The .ncbirc file should always start with the line (square brackets included):

```
[NCBI]
```

And then three more lines:

```
Data=/your/path/here
[BLAST]
BLASTDB=/your/database/path/here
```

where "/your/path/here" is the path to the directory where you store the BLAST matrix and data files, and "/your/database/path/here" is

the path to the directory containing the BLAST databases.

Example:
You have the data directory with matrix files in /usr/local/blast/ and want to put all your BLAST databases in the directory "databases" located in: /usr/local/, then the .ncbirc file should contain these four lines:

```
[NCBI]
Data=/usr/local/blast/data
[BLAST]
BLASTDB=/usr/local/databases
```

Preformatted BLAST databases ready to use can be downloaded from the ncbi ftp server at:
`ftp://ftp.ncbi.nih.gov/blast/db/`

These databases need to be unzipped, untarred and placed in the directory you have specified in the .ncbirc file.
Fasta files for some of the most common databases that can be "BLAST-formatted " are also located at the ncbi ftp site:
`ftp://ftp.ncbi.nih.gov/blast/db/FASTA`

Any Fasta formatted sequence file can be turned into a BLAST database. In the next paragraph of this tutorial we will explain how you can create your own custom databases.

Formatdb

To create your own BLAST databases you must use a program called formatdb.
There are other tools available, but for this tutorial we will use formatdb. Formatdb can index Fasta formatted "flatfiles" containing protein or nucleotide sequences, the program was preinstalled with the ncbi-tools package.

formatdb is run from the command line (terminal) using the following important arguments:

```
-t Title for the database file
-i Input file for formatting
-p Type of file
   T - protein
   F - nucleotide
```

Example:
To create a BLAST database called "mydatabase" using nucleotide sequences from a file called `drosoph.nt` (please download the file from the ncbi ftp site or use one of your own, you can also download it from the tutorials at www.ebioinformatics.org) write the following on the terminal window (you must be in the folder containing the file `drosoph.nt`!):

```
$sudo /usr/local/blast/bin/formatdb -p F -t
mydatabase -i drosoph.nt
```

You should then obtain in that folder a number of files called `drosoph.nt` having different file suffixes like `.nhr .nin .nnd .nni .nsd .nsi` and `.nsq`. These files should be moved to your blast database directory described above. If you format a file containing protein sequences the file suffixes should be `.phr .pin .pnd .pni .psi` and `.psq` (the starting n(ucleotide) exchanged for p(rotein).
To find out more about formatdb look in the README's located in /usr/local/blast (if you have installed the ncbi-tools of course).

Creating a web interface to BLAST: WWWBLAST

This package is based on the standalone WWW BLAST server suite of programs created by the NCBI team.
First you must install our package "wwwblast.pkg" as usual (see Figure 1 on next page).

After this automatic installation you should have a folder containing the web interface to BLAST in /Library/WebServer/Documents/blast.

In the next step you must change some parameters in the Apache web-server installation to allow the execution of the installed blast scripts. This is done by editing some configuration files.

A -First you have to change your web-server parameters. You do this by editing a file called httpd.conf. We will use a text-editor called "pico", do like this (**$** is only to symbolize the prompt):

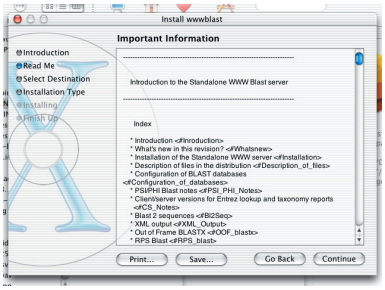
```
$sudo pico /etc/httpd/httpd.conf
```

First modify the Options line adding ExecCGI where the file looks like this:

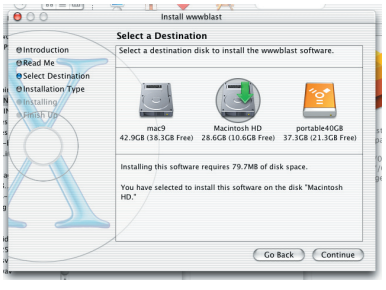
```
# This may also be "None", "All", or any
combination of "Indexes",
# "Includes", "FollowSymLinks", "ExecCGI", or
"MultiViews".
#
# Note that "MultiViews" must be named
*explicitly* --- "Options All"
# doesn't give it to you.
#
    Options Indexes FollowSymLinks MultiViews
```

Change it to this: (the modification is shown in red)

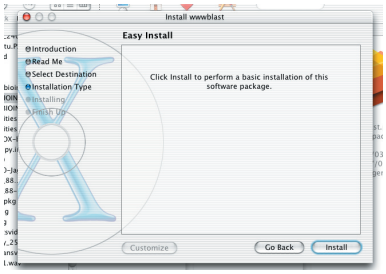
Step 1 Please start the install procedure as usual. Double-click on the unpacked installer named wwwblast.pkg. You will see here that the files will be installed on the Webserver that comes with MacOSX. MacOSX uses Apache as webserver.



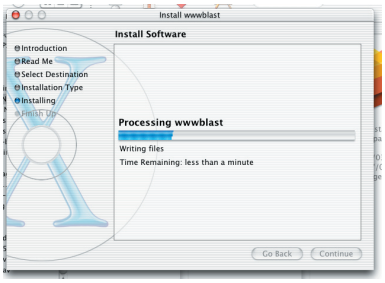
Step 2 Please read carefully this information from the NCBI team. You will find the same information later on as a web page on the installed wwwblast.



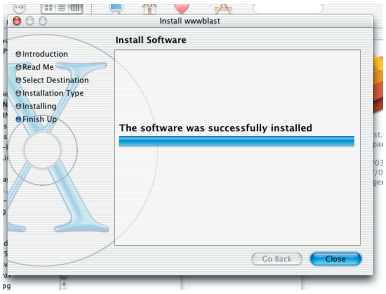
Step 3 Now you must select the harddisk where you have your WebServer, usually the same as the disk containing the System Folder.



Step 4 This is the last step before the install starts. Click on install.



Step 5 Now you can relax and the process will start, this takes a few minutes



Step 6 Now the automatic part is finished and we start with some manual operations to adjust your system.

Figure 1 Installation of the BLAST package

```
# This may also be “None”, “All”, or any
combination of “Indexes”,
# “Includes”, “FollowSymLinks”, “ExecCGI”, or
“MultiViews”.
#
#  Note that “MultiViews” must be named
*explicitly* --- “Options All”
# doesn’t give it to you.
#
  Options Indexes FollowSymLinks MultiViews ExecCGI
```

B - then find a line that looks like this:

```
# To use CGI scripts:
#
#AddHandler cgi-script .cgi
```

Uncomment (erase) the # before AddHandler and add '.pl', like this:

```
# To use CGI scripts:
#
 AddHandler cgi-script .cgi .pl
```

After editing press control-X , say yes, return and quit pico. (fig 2.)

C - Now you must add a file called `.htaccess` (please pay attention that the name starts with a dot, dots at the beginning of a name makes the files "invisible" on Unix systems) in the same folder that our package installed the wwwblast (/Library/WebServer/Documents/blast). We use the program pico again.

A session in pico looks like this:

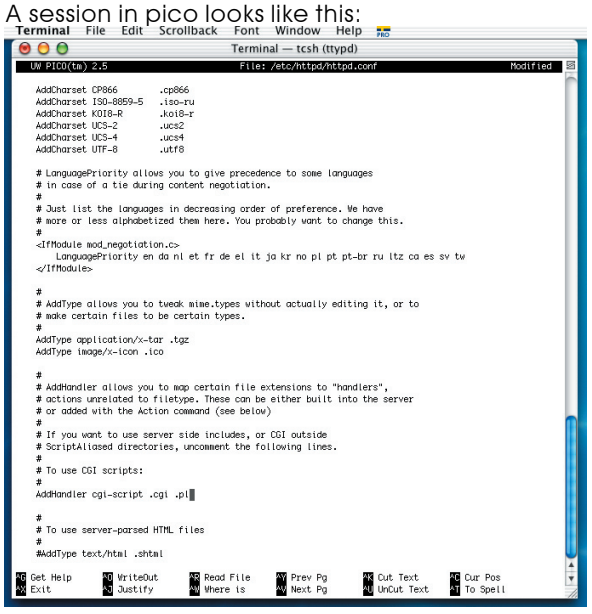


Figure 2. Shows pico when we edited the httpd.conf file. Please remember that this text editor is not like word and you can not use the mouse to navigate, But you avoid having format problems later on!!

```
$sudo pico /Library/Webserver/Documents/blast/.htaccess
```

When pico starts add the following line at the end of the file:

```
Options +ExecCGI
```

Then control-X, answer yes, return and it is done. Restart your web server in System preferences (go to Sharing, select Personal Web Sharing and click Start or Stop button once or twice depending of the current state of your web server):

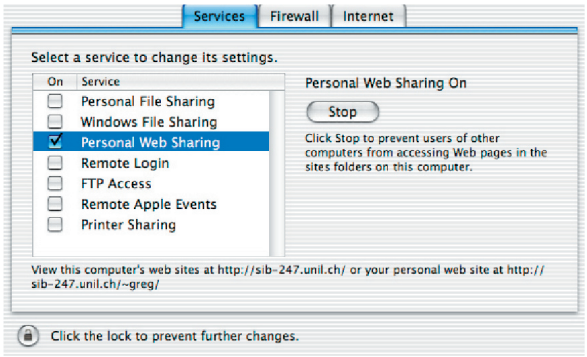


Figure 3. How to restart you web server

Now you are the lucky owner of a workstation/server running your very own blast-server.

To test your installation open your favourite web-browser and type http://localhost/blast (from another computer change localhost with the URL of that machine e.g., http://mymachine.slu.se) you should see a page like this:

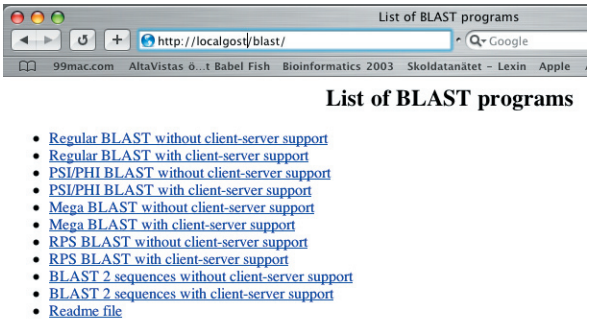


Figure 4. The picture illustrates just an example of layout you can configure this in the way that suites you. This are html for template purposes only.

How to connect your databases to the web-interface?

First install your blast-formatted databases in your selected folder as described above. Secondly you must tell the web-interface where to find your databases, this is done by editing some files called: `blast.rc`, `psiblast.rc` and the corresponding html files. Use the perl script that is supplied by the NCBI team (`config_setup.pl`).
This perl script is located in your www-blast folder (/Library/WebServer/Documents/blast/).
First we move the test databases to a new directory and subsequently we create a link "db" to the real directory:

```
$sudo mv db dbbak
$sudo ln -s /usr/local/databases db
```

Then you should run the perl script, in this case we decided to install our databases in /usr/local/databases therefore the command is:

```
$sudo perl config_setup.pl /usr/local/databases out
```

where /usr/local/databases is the place where we keep the blast databases and "out" is a temporary directory where we can find the files created use them to replace the original ones.

```
$sudo cp out/blast.rc .
$sudo cp out/psiblast.rc .
```

Please pay attention to the ".”! (it means the current directory)
Now you have to edit the html pages to show your new databases.

```
$sudo pico blast.html
```

It should look like this:

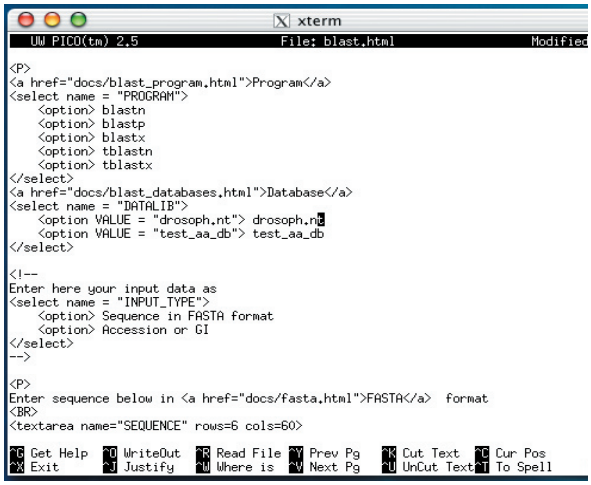


Figure 5. Edition of the blast.html file with Pico

If you have done no mistakes you will be ready to try your first search. Start your web browser and go to your wwwblast pages (Figure 6). Paste a query sequence, select the parameters and click "search" button. After a while you should get the result page (Figure 7).

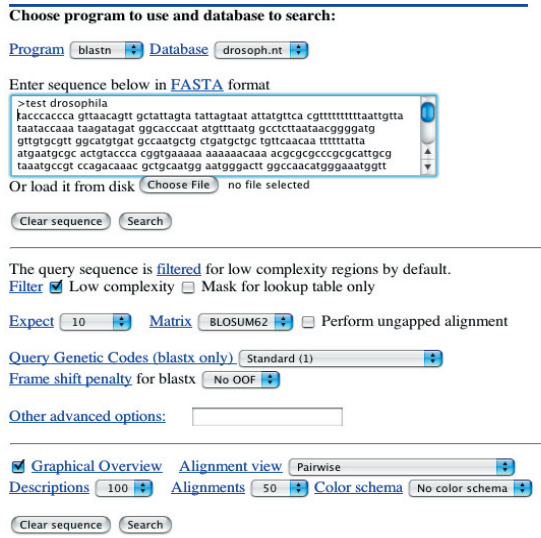


Figure 6. In this example we have the test database we formatted "drosoph.nt" (you can download this file from the tutorial pages at www.ebioinformatics.org).
There are also two test blast databases on this installation test _ na _ db (nucleic) and test _ aa _ db (protein).

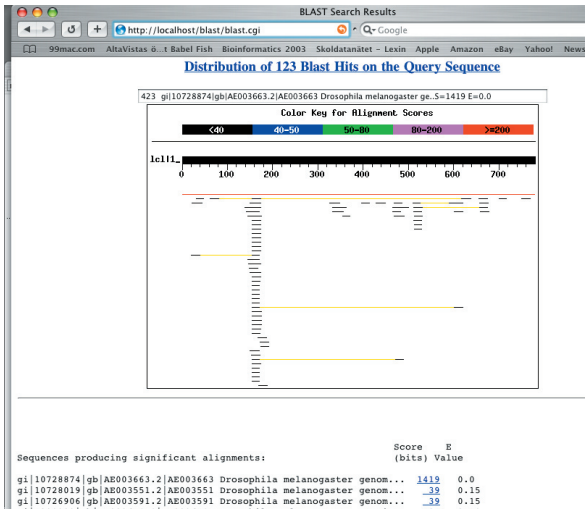


Figure 7. The result of a simple search using a nucleic acid sequence against the Drosophila genomic database.

Now you have a working web interface to the blast programs. Install your databases and connect them as explained. Lycka till! as we say in Sweden.

Installing a GUI interface to EMBOSS

There are several graphical user interfaces (GUI) and Web interfaces for the EMBOSS package. Please visit the EMBOSS web site to read more about this. In this second tutorial we have chosen to install Thomas Siegmund`s creation. He uses "Kaptain" created by Terék Zsolt to wrap GUIs to the EMBOSS applications. Kaptain is a universally applicable graphical front-end based on context-free grammars. You need EMBOSS (installed in our first part), Kaptain and QT-libraries installed. Our packages contain:

a- QT installation package (install it first)
b- Kaptain installation package (install it next)
c- Emboss _ kaptain package (install it last)

If all the steps are made correctly you can start all EMBOSS programs from an X-Windows terminal with a module called "embosslauncher.kaptn" you start the GUI like this:

```
$ embosslauncher kaptn
```

You should get a window like this one:

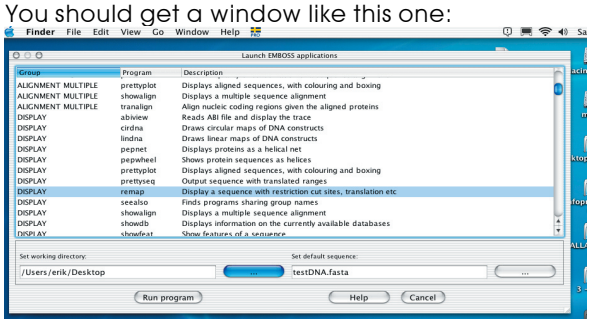


Figure 8. Shows the central command in Kaptain, you can start all EMBOSS programs from here.

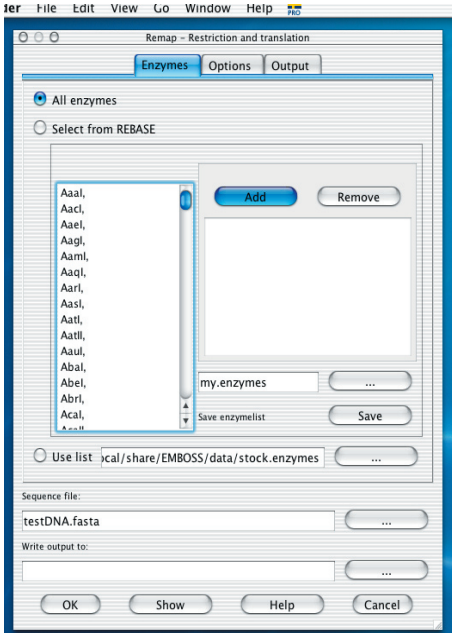


Figure 9. Here you see Kaptain running remap. OBS:You can also start all programs individually.

Most configurations are done automatically, but if you encounter any problem, read the Problems note (see Addendum). For up to date changes please visit `http://www.ebioinformatics.org`

Use "showdb" to check that it worked
`$showdb.kpt`

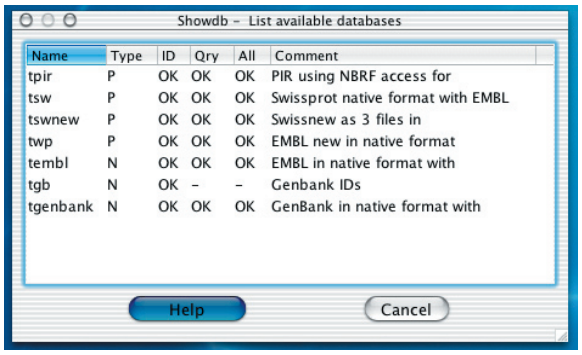


Figure 10. You should now get a list of the installed databases. You can later adapt the emboss.default to reflect your database installation

Nedit

The most interesting feature with this EMBOSS GUI is the incorporation of a sequence editor.
For the preview of text and sequence data and for the display of error messages Thomas Siegmund uses Nedit. Nedit is a fast, powerful and lightweight open source text editor. We have created a new Fink package including Nedit in the installation, please install it. You can of course install the binaries downloading directly from Nedits home page (See web resources).

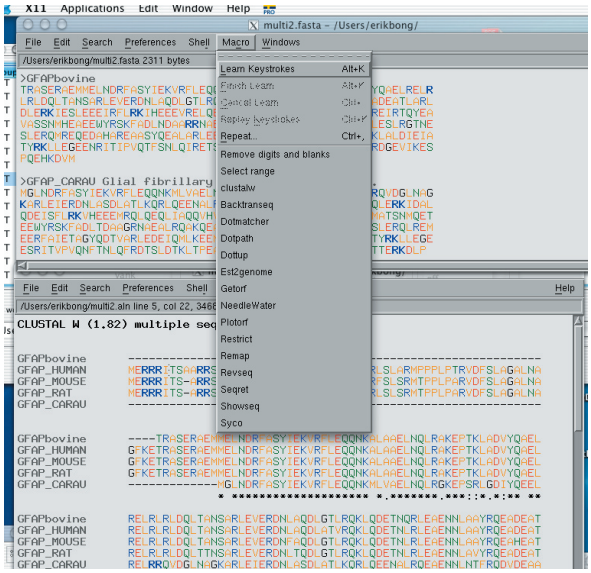


Figure 11. In this figure you see Nedit with a multi fasta protein list. From Nedit we launched ClustalW and we obtained the result on another fully editable window.

Thomas created also a `.nedit` settings file. Placing this `.nedit` file in your home directory allows Nedit to show some common sequence file formats with a beautiful highlighted syntax. Please download his file from
`http://userpage.fu-berlin.de/~sgmd/.nedit`
Installing the parameters mentioned above enables to start a nice ClustalW alignment from Nedit.

Start nedit from your X-Windows terminal (fig. 11).
`$nedit`

Extra packages: ClustalW and T-Coffee

To work with EMBOSS.kaptain and other future package installations, we also recommend to install the following packages in the section "New-packages": ClustalW and T _ Coffee, visit: `http://www.ebioinformatics.org` to download them.
This two packages work on the command line without any special configuration and ClustalW works also through other installed GUI interfaces, eg. Emma and Kaptain. Do exactly in the same way as we explained in tutorial part I (see EMBnetNews vol, 9 nr 1: `http://www.embnet.org/download/embnetnews/index.html`).
ClustalW is is a general purpose multiple sequence alignment program for DNA or proteins.
T_Coffee is also a multiple sequence alignment package with some extra features: e.g., it allows you to combine results obtained with several alignment methods. For instance if you have an alignment coming from ClustalW, and another alignment coming from Dialign, T-Coffee will process the information and produce a new consensus multiple sequence alignment using the incoming data.

Next Part?

In part 3 we are going to explain how to install Jemboss and a web-interface to EMBOSS. We will also install a GUI for T _ Coffee.
For all of you wondering why MacOSX and bioinformatics: we will also review the new 64-bit G5 Macintosh machines and do tests to compare Altivec-enhanced programs and the same programs running on other platforms.
Do not miss it!

Web addresses

http://www.embnet.org
http://www.ebioinformatics.org
http://www.ncbi.nlm.nih.gov/
http://www.emboss.org
http://userpage.fu-berlin.de/~sgmd/
http://www.trolltech.com/
http://kaptain.sourceforge.net/
http://www.nedit.org/
http://www.ebi.ac.uk/clustalw/

http://igs-server.cnrs-mrs.fr/~cnotred/Projects _ home _ page/t _ coffee _ home _ page.html

References

1 Altschul S.F., Madden T.L., Schäffer A.A., Zhang J., Zhang Z., Miller W., Lipman D.J. (1997)
Gapped BLAST and PSI-BLAST: a new generation of protein database search programs.
Nucleic Acids Res. 25, 3389-3402.

2 Rice P., Longden I., Bleasby A. (2000)
EMBOSS: The European Molecular Biology Open Software Suite
Trends in Genetics, 16, 276-277.

3 Thompson J.D., Higgins D.G., Gibson T.J. (1994)
CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice.
Nucleic Acids Res. 22:4673-4680.

4 Notredame C., Higgins D., Heringa J. (2000)
T-Coffee: A novel method for multiple sequence alignments.
Journal of Molecular Biology 302, 205-217.

5 Bongcam-Rudloff E. (2003)
A guide on how to install Bioinformatics tools on MacOSX (Part1).
Embnetnews, vol.9 nr.1, 13-16.

Glossary

sudo This command means "superuser do". It enables allowed users to execute a single command with the root user priviledges. You must have administrator status to invoke the sudo command. This command is used to access permissions and ownerships or perform tasks reserved for root. To be used with caution!

pico Pico is a simple, display-oriented, text editor based on the Pine message system composer. As with Pine, commands are displayed at the bottom of the screen, and context-sensitive help is provided.
As characters are typed they are immediately inserted into the text.

QT Qt is a multiplatform, C++ application development framework created by Trolltech, Norway. Qt lets developers write a single application source that will run natively on Linux/Unix, Windows, Mac OS X, and embedded Linux with a simple recompile.

unzipped From unzip. Archives compressed in zip format can be unpacked using the command unzip. Normal command: `$unzip myfile.zip`

untarred From tar. The tar command creates, adds files to, or extracts files from an archive file in tar format.
Normal command: `$tar –xvf yourfile.tar`
You can also use the free Stuffit Expander, Stuffit unzip and untar files to a folder automatically.

Addendum

Most configurations are done automatically, but if you encounter any problem try this:

1 To instruct some of the Kaptain-EMBOSS programs where you keep your data files, like restriction enzymes, you must change your environment.
In our first tutorial (part I) you created a file in your home directory called «.cshrc».

```
source /sw/bin/init.csh
setenv PLPLOT_LIB /usr/local/share/emboss
set path=( /usr/local/bin /usr/local/blast/bin
/usr/local/staden/macosx-bin ${path} )
setenv DISPLAY :0.0
```

Add the following line to that file (use pico!):

```
setenv EMBOSS_DATA /usr/local/share/EMBOSS/data
```

so that it looks like this:

```
source /sw/bin/init.csh
setenv PLPLOT_LIB /usr/local/share/emboss
setenv EMBOSS_DATA /usr/local/share/EMBOSS/data
set path=( /usr/local/bin /usr/local/blast/bin
/usr/local/staden/macosx-bin ${path} )
setenv DISPLAY :0.0
```

2 To get access to sequence databases connected to EMBOSS you have to edit a file called emboss.default. The normal installation installs a template. Now we are going to change it to reflect our installation. You should adapt this file when you install your own databases, for the moment we are going to use some preinstalled test databases.

```
$cd /usr/local/share/EMBOSS
$sudo cp emboss.default.template emboss.default
$sudo pico emboss.default
```

Change a line that looks like this:

```
#SET emboss_tempdata path_to_directory_
$EMBOSS/test
```

remove the # and change the path to:

```
SET emboss_tempdata /usr/local/share/EMBOSS/test
```

Use "showdb" to check that it worked

```
$showdb.kpt
```

THE PHYSIOLOGICAL HOLY GRAIL?

By Vivienne Baillie Gerritsen

Though we seem to be quite solid, we are in fact quite liquid. The best part of us is water. In fact, all living creatures – from bacteria to plants and man – are made up of roughly 70% water. And that water needs to flow into us, out of us and inside us. We sweat water, we cry water, we digest with water, we think thanks to water, we pee water – to name but a few aqueous physiological activities. Imagine that, on a daily basis, hundreds of litres of water go through a human kidney! It has been known for decades that water molecules can quite happily cross cell membranes unassisted. However, transit in this way could not account for the huge amounts a kidney would need for example. There needs to be another system. In the 1990s such a system was discovered: aquaporin. Aquaporins are proteins which are embedded within cellular or intracellular membranes and function as a rather high-tech channel specific to water molecules. What is more, aquaporins have been found in plants, animals and yes, bacteria.

In the 1950s, scientists were already wondering how on earth huge amounts of water could possibly diffuse through membrane lipid bilayers. Diffusion in this way is indeed effective but rather slow and simply could not account for it. Surely there must exist some kind of water pore. It took a further thirty years before the very first aquaporin was actually discovered…and quite by chance. Indeed, scientists had been rummaging around red blood cells on the lookout for a specific Rhesus factor molecule when they stumbled upon an odd protein lodged in the plasma membrane. After close inspection, it turned out that this protein was in fact a pore which let water in and out of the cell. Aquaporin has been called the Holy Grail of fluid-transport physiology….

What could a physiological Holy Grail possibly look like? Well…a dumbbell. Or to be more precise an assembly of four dumbbells. Indeed, one aquaporin channel is made up of four aquaporin monomers, each of which acts as a specific water pore. An aquaporin monomer has a diameter of about 30Å and a height of about 60Å. The four monomers are quite tightly bound and form a stable complex in the plasma membrane. Were you to unravel one, you would find six longish alpha helices and two shortish ones. Imagine a cylinder. Take six tubes and place them vertically around the cylinder. That is the way the six helices wrap themselves around the centre. Tie a ribbon around the centre of the pore. It forms a constriction in the middle, of a diameter of about 8Å. This is what gives the dumbbell shape to the aquaporin monomers – though there is no ribbon, simply molecular forces working away. The two short helices tuck themselves into the very middle of the pore and form in effect a barrier to molecules other than water. These two short helices are what make an aquaporin what it is, i.e. a pore for water molecules only.

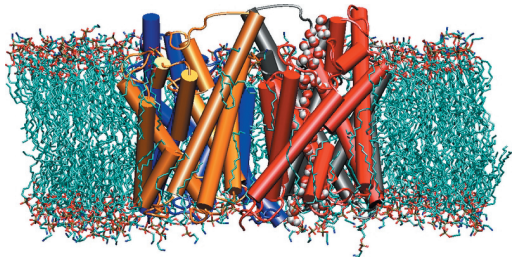


Fig. 1 3D model of aquaporin

Source: http://www.psc.edu/science/2002/schulten/precious_bodily_fluids.

How is water driven through an aquaporin then? The passage of water molecules through the pores is rather charming and has been described as a molecular ballet. Nothing complicated, no back-breaking arabesque or hurried pas de deux, no just a smooth glide and then a graceful pirouette. What happens is that these aqueous ballerinas drift into an aquaporin one by one, face down so to speak, i.e. their oxygen facing the inside of the pore. Though perhaps 'drift' is not quite the word since one billion molecules cross a membrane…per second! However, drift or no drift, once they reach the middle of the channel, the water molecules are grabbed by side chains which line the interior of the pore and swung around so that they exit the pore, bottom down so to speak, i.e. hydrogens first.

What on earth would they do this for? Well the system is ingenious. It all has to do with

protons. Indeed, not only do aquaporins let water through but they also inhibit the transit of protons. And this is a very wise move. Why? Protons are needed to charge cells; if cells lose their protons, they lose their energy. The thing is, protons usually hitch a ride on the backs of water molecules. A line of water molecules forms what is known as a proton wire, a path on which protons can move from one place to another. Now if this path is disrupted, the protons have only one choice but to go back from where they came. This is precisely what happens in an aquaporin. And thanks to the two short helices mentioned earlier. It is precisely at this point that the incoming water molecules perform a pirouette and disrupt the proton wire. The protons go back home while the water molecules continue their way to the other side. Besides, the two short helices, there is also a second major mechanism that stops other molecules from passing through: the constriction. The size of the constriction is such that it only lets molecules the size of water molecules through…

Evidently, aquaporins must be at the heart of a number of diseases if their role is, as is the role of water, so important. That is why it is crucial to get to know aquaporins on a very intimate level, with a view to develop new therapeutics thanks to the world of drug design. As an example, corneal transparency, i.e. vision, requires a precise regulation of water content. Indeed, it could be that water has a role in keeping the diameter and spacing of collagen regular, which in turn confers transparency to the lens. Cataracts are a direct consequence of aquaporin malfunction. Nephrogenic diabetes insipidus is a disorder in which patients' kidneys cannot reabsorb water correctly; patients have to go to the bathroom frequently and end up dehydrated. Defective aquaporins are no doubt at the heart of this form of disease. Aquaporin malfunctions in salivary and lacrimal glands also result in a disorder known as Sjogrens syndrome, or dry mouth. Besides the malfunction of aquaporins, their overexpression can be indicative of physiological states such as congestive heart failure or pregnancy. Though the latter should not be a cause to worry. No doubt, within the next decade, a growing number of clinical disorders will point their finger at a deregulation of aquaporin function or expression. One interesting biotechnological development could be to incorporate aquaporins to certain materials which could then filter ions, such as salt, from seawater and produce fresh water in countries where it is so scarce.

Cross-references to Swiss-Prot

P29972: human aquaporin 1
P43285: *Arabidopsis thaliana* (mouse ear-cress) aquaporin PIP1.1
Q9C4Z5: *Methanobacterium* thermoautotrophicum aquaporin aqpM
Q23808: *Cicadella viridis* (Green leafhopper) aquaporin AQPcic

References

1. Tajkhorshid E., Nollert P., Jensen M.Ø., Miercke L.J.W. , O'Connell J., Stroud R.M., Schulten K.
Control of the selectivity of the aquaporin water channel family by global orientational tuning
Science 296:525-30(2002).
PMID: 11964478

2. Kozono D., Yasui M., King L.S., Agre P.
Aquaporin water channels: atomic structure and molecular dynamics meet clinical medicine
J. Clin. Invest. 109:1395-9(2002).
PMID: 12045251

3. Structure, Dynamics, and function of aquaporins
NIH resource for macromolecular modeling and bioinformatics
http://www.ks.uiuc.edu/Research/aquaporins/

Protein Spotlight (ISSN 1424-4721) is published by the SWISS-PROT group at the Swiss Institute of Bioinformatics (SIB).

National Nodes

Argentina
Oscar Grau
IBBM, Facultad de Cs. Exactas, Universidad Nacional de La Plata
Email: grau@biol.unlp.edu.ar
Tel: +54-221-4259223 Fax: +54-221-4259223
http://www.ar.embnet.org

Australia
Sonia Cattley
RMC Gunn Building B19, University of Sydney,NSW, 2006
Email: scattley@angis.org.au
Tel: +61-2-9531 2948
http://www.au.embnet.org

Austria
Martin Grabner
Vienna Bio Center, University of Vienna
Email: martin.grabner@univie.ac.at
Tel: +43-1-4277/14141
http://www.at.embnet.org

Belgium
Robert Herzog, Marc Colet
BEN ULB Campus Plaine CP 257
Email: rherzog@ulb.ac.be mcolet@ulb.ac.be
Tel: +32 2 6505146 Fax: +32 2 6505124
http://www.be.embnet.org

Brasil
Gonçalo Guimaraes Pereira
Laboratório de Genômica e Expressão - IB
UNICAMP-CP 6109
13083-970 Campinas-SP, BRASIL
Tel: 0055-19-37886237/6238
Fax: 0055-19-37886235
Email: goncalo@unicamp.br
http://www.lge.ibi.unicamp.br

Canada
Laura Brown
Canadian Bioinformatics Resource, National Research Council Canada, Institute for Marine Biosciences,
Email: manager@cbr.nrc.ca
Tel: +1-902-426 7310 Fax: +1-902-426 9413
http://www.ca.embnet.org

Chile
Dr. Ricardo Baeza-Yates
Dept. of Computer Science, Santiago,
Email: rbaeza@dcc.uchile.cl
http://www.embnet.cl

China
Jingchu Luo
Room 303, Exchange Centre, Peking University
Email: luojc@cbi.pku.edu.cn
Tel: +86-10-6275 9001
http://www.cbi.pku.edu.cn

Colombia
Emiliano Barreto Hernández
Instituto de Biotecnología
Universidad Nacional de Colombia
Edificio Manuel Ancizar
Bogota - Colombia
Tel: +571 3165027 Fax: +571 3165415
Email : ebarreto@ibun.unal.edu.co
http://bioinf.ibun.unal.edu.co

Cuba
Ricardo Bringas
Centro de Ingeniería Genética y Biotecnología, La Habana,
Email: bringas@cigb.edu.cu
Tel: +53 7 218200
http://www.cu.embnet.org

Denmark
Hans Ullitz-Moeller
BioBase, University of Aarhus
Email: hum@biobase.dk
Tel: +45-86-13 9788
http://www.dk.embnet.org

Finland
Kimmo Mattila
CSC, Espoo
Email: Kimmo.Mattila@csc.fi
Tel: +358 9 457 2708
http://www.fi.embnet.org

France
PLAZA Jean-Marc
INFOBIOGEN, Evry
Email: plaza@infobiogen.fr
Tel: +33 1 60 87 37 11 Fax: +33 1 60 87 37 96
http://www.fr.embnet.org

Germany
Sandor Suhai
EMBnet node at the German Cancer Research Center
Department of Molecular Biophysics (H0200)
Email: genome@dkfz.de
Tel: +49-6221-422 342 Fax: +49-6221-422 333
http://www.de.embnet.org

Greece
Babis Savakis
Institute of Molecular Biology and Biotechnology
Heraklion, Crete
Email: savakis@nefeli.imbb.forth.gr
Tel: +30-81-391 114 Fax: +30-81-391 104
http://www.imbb.forth.gr

Hungary
Endre Barta
Agricultural Biotechnology Center
Szent-Gyorgyi A. ut 4. Godollo,
Email: barta@abc.hu
Tel: +36 30-2101795
http://www.hu.embnet.org

India
H.A.Nagarajaram
Laboratory of Computational Biology & Bioinformatics facility, Centre for DNA Fingerprinting and Diagnostics (CDFD), Hyderabad
Email: han@www.cdfd.org.in
Tel: +91 40 7155607 / 7151344 ext:1206
Fax : +9140 7155479
http://www.in.embnet.org

Israel
Leon Esterman
INN (Israeli National Node) Weizmann Institute of Science
Department of Biological Services, Biological Computing Unit, Rehovot
Email: Leon.Esterman@weizmann.ac.il
Tel: +972- 8-934 3456
http://www.il.embnet.org

Italy
Cecilia Saccone
CNR - Institute of Biomedical Technologies
Bioinformatics and Genomic Group
Via Amendola 168/5 - 70126 Bari (Italy)
Email: saccone@area.ba.cnr.it
Tel. +39-80-5482100 - Fax. +39-80-5482607
http://www.it.embnet.org

Mexico
Cesar Bonavides
Nodo Nacional EMBnet, Centro de Investigación sobre Fijación de Nitrógeno, Cuernavaca, Morelos
Email: embnetmx@cifn.unam.mx
Tel: +52 (7) 3 132063
http://embnet.cifn.unam.mx

The Netherlands
Jack A.M. Leunissen
Dept. of Genome Informatics
Wageningen UR, Dreijenlaan 3
6703 HA Wageningen, NL
Email: Jack.Leunissen@wur.nl
Tel: +31 317 484074
http://www.nl.embnet.org

Norway
Rune Groven
The Norwegian EMBnet Node
The Biotechnology Centre of Oslo
Email: admin@embnet.uio.no
Tel: +47 22 84 0535
http://www.no.embnet.org

Poland
Piotr Zielenkiwicz
Institute of Biochemistry and Biophysics
Polish Academy of Sciences Warszawa
Email: piotr@pl.embnet.org
Tel: +48-22 86584703
http://www.pl.embnet.org

Portugal
Pedro Fernandes
Instituto Gulbenkian de Ciencia
Unidade de Bioinformatica
2781-901 OEIRAS
Email: pfern@igc.gulbenkian.pt
Tel: +351 214407912 Fax: +351 2144079070
http://www.pt.embnet.org

Russia
Sergei Spirin
Biocomputing Group, Belozersky Institute Moscow
Email: sas@genebee.msu.su
Tel: +7-095-9328825
http://www.genebee.msu.su

Slovakia
Lubos Klucar
Institute of Molecular Biology SAS Bratislava
Email: klucar@embnet.sk
Tel: +421 7 5941 2284
http://www.sk.embnet.org

South Africa
Ruediger Braeuning
SANBI, University of the Western Cape, Bellville
Email: ruediger@sanbi.ac.za
Tel: +27 (0)21 9593645
http://www.za.embnet.org

Spain
José M. Carazo, José R. Valverde
EMBnet/CNB, Centro Nacional de Biotecnología, Madrid
Email: carazo@es.embnet.org, jrvalverde@es.embnet.org
Tel: +34 915 854 505 Fax: +34 915 854 506
http://www.es.embnet.org

Sweden
Nils-Einar Eriksson, Erik Bongcam-Rudloff
Uppsala Biomedical Centre, Computing Department, Uppsala, Sweden
Email: nils-einar.eriksson@bmc.uu.se
erik.bongcam@bmc.uu.se
Tel: +46-(0)18-4714017, +46-(0)18-4714525
http://www.embnet.se

Switzerland
Laurent Falquet
Swiss Institute of Bioinformatics, CH-1066 Epalinges/ Lausanne
Email: Laurent.Falquet@isb-sib.ch
Tel: +41 (21) 692 5954 Fax: +41 (21) 692 5945
http://www.ch.embnet.org

United Kingdom
Alan Bleasby
UK MRC HGMP Resource Centre, Hinxton, Cambridge
Email: ableasby@embnet.org
Tel: +44 (0) 1223 494535
http://www.uk.embnet.org

Specialist Nodes

EBI
Rodrigo López
EBI Embl Outstation, Wellcome trust Genome Campus
Hinxton Hall, Hinxton, Cambridge, United Kingdom
Email: rls@ebi.ac.uk
Phone: +44 (0)1223 494423
http://www.ebi.ac.uk

ETI
P.O. Box 94766
NL-1090 GT Amsterdam, The Netherlands
Email: wouter@eti.uva.nl
Phone: +31-20-5257239
Fax: +31-20-5257238
http://www.eti.uva.nl

EU
Dr Bernard Mulligan
DG Research - European Commission
Brussels BELGIUM
Email: bernard.mulligan@cec.eu.int

ICGEB
Sćndor Pongor
International Centre for Genetic Engineering and Biotechnology
AREA Science Park, Trieste, ITALY
Email: pongor@icgeb.trieste.it
Phone: +39 040 3757300
http://www.icgeb.trieste.it

LION Bioscience
Peter Rice
LION Bioscience AG, Heidelberg, Germany
Email: Peter.Rice@uk.lionbioscience.com
Phone: +44 1223 224700
http://www.lionbioscience.com

MIPS
H. Werner Mewes
Email: mewes@mips.embnet.org
Phone: +49-89-8578 2656
Fax: +49-89-8578 2655
http://www.mips.biochem.mpg.de

Pharmacia - Biovitrum
Timothy Wood
Biovitrum, Stockholm, Sweden
Email: timothy.wood@eu.pnu.com
Phone: +46 (8) 695 9134
http://www.pharmacia.se

Sanger Institute
Michelle Clamp
Welcome Trust Sanger Institute, Wellcome Trust Genome
Campus, Hinxton Hall, Hinxton CB10 1SD, Cambridge, United Kingdom
Email: michele@sanger.ac.uk
Phone: +44-1223-494 967
Fax: +44-1223-494 919
http://www.sanger.ac.uk

UMBER
Terri Attwood
School of Biological Sciences, The University of Manchester
Oxford Road, Manchester M13 9PT, UK
Email: attwood@bioinf.man.ac.uk
Phone: +44 (0)61 275 5766
Fax: +44 (0) 61 275 5082
http://www.bioinf.man.ac.uk/dbbrowser

TECH-MGR
Email: tech-mgr@embnet.org
The team gives support to EMBnet nodes and helps them with maintenance and troubleshooting.
The team is formed of experienced system administrators and programmers who ensure the availability of local services for all EMBnet users.

EMBnet.news
ISSN 1023-4144

Dear reader,

If you have any comments or suggestions regarding this newsletter we would be very glad to hear from you. If you have a tip you feel we can print then please let us know. Please send your contributions to one of the editors. You may also submit material by Internet E-mail.

Past issues of embnet.news are available as PostScript or PDF files (ISSN 1023-4144). You can get them by anonymous ftp from:
the EMBnet organisation Web site
http://www.embnet.org/download/embnetnews
the Belgian EMBnet node
ftp://ftp.be.embnet.org/pub/embnet.news
the UK EMBnet node
ftp://ftp.uk.embnet.org/pub/embnet.news
the EBI EMBnet node
ftp://ftp.ebi.ac.uk/pub/embnet.news

Submission deadline for next issues:

October 31, 2003
February 29, 2004

EMBnet.news is an official publication of the EMBnet organisation
www.embnet.org