Installation and Usage of MultiFinder, SequenceExtractor and BlockFilter

I. Introduction:

MultiFinder is a tool designed to combine the results of multiple motif finders and analyze the resulting motifs with a unified statistical framework. MultiFinder was designed to utilize DNA sequence conservation between organisms as a method of enriching for functionally conserved sequence elements. A tool for extracting conserved sequence was implemented called SequenceExtractor.pl. This tool allows the extraction of conserved regions of sequence from the human genome using available alignments with other genomes. SequenceExtractor.pl generates input sequence files for both the query set and the background that can be used directly with MultiFinder.

II. Checklist of Software and Other Files Necessary for MultiFinder, SequenceExtractor, and BlockFilter:

Software:

Perl
XML::Writer module for Perl
AlignACE
BioProspector
MDscan
MEME
ScanACE
CompareACE
MotifStats
Tree
List_clusters.pl

Genomic Sequence and Anotation Files:

chr1.fa through chrY.fa
chr1.axt through chrY.axt
Hs.seq.uniq.gz
refGene.txt

MultiFinder Programs and Modules:

AnalysisFunctions.pm
MotifFunctions.pm
MotifObject.pm
ScanFunctions.pm
ScanObject.pm
SequenceFunctions.pm
BlockFilter.pl
MultiFinder.pl
SequenceExtractor.pl

III. Installation Instructions:

Required Supporting Programs for MultiFinder:
MultiFinder can currently call four motif finders (AlignACE [1,2], BioProspector [3], MDscan [4], and MEME [5]) that were developed previously by other groups. These programs can either be installed and called from a command line or have their absolute path specified in the Configure.ini file. The Configure.ini file tells MultiFinder where to find the motif finders and supporting programs.

AlignACE [1,2]
http://atlas.med.harvard.edu/
The distribution for AlignACE contains both AlignACE and CompareACE in a precompiled format along with the source code for the programs. Once the alignace2004.tar file has been downloaded the file can then be unpacked using the command:

```
tar –xvf alignace2004.tar
```

This will generate a directory containing the executables and source programs for AlignACE and CompareACE. These programs must either be placed in a directory specified in the user's PATH variable or the absolute path must be specified in the Configure.ini file.

BioProspector [3]
http://ai.stanford.edu/~xsliu/BioProspector/
The BioProspector distribution comes as a zip file that can be unpacked using the unzip command:

```
unzip Bioprospector1.Feb03.zip
```

This will generate a number of versions of the BioProspector program for different operating systems. The version that is applicable to the users system should be selected. The name of the program should be changed from BioProspector.<operating system> to just BioProspector. If the program is not yet executable, the user will need to change the permissions as follows:

```
mv BioProspector.linux BioProspector
chmod u+x BioProspector
```

The resulting executable file should be placed in a directory specified in the users PATH variable or the absolute path should be specified in the Configure.ini file.

MDscan [4]
http://ai.stanford.edu/~xsliu/MDscan/
The MDscan distribution comes as a zip file that can be unpacked using the unzip command:

```
unzip MDscan1.May03.zip
```

This will generate a number of versions of the MDscan program for different operating systems. The version that is applicable to the users system should be selected. The name of the program should be changed from MDscan.<operating system> to just MDscan. If the program is not yet executable, the user will need to change the permissions:

```
mv MDscan.linux MDscan
```

```
chmod u+x MDscan
```

The resulting executable file should be placed in a directory specified in the user's PATH variable or the absolute path should be specified in the Configure.ini file.

MEME [5]
http://meme.sdsc.edu/meme/meme-download.html
The installation of the MEME program is more involved and is the only program here that requires a true full installation. This should be done by the administrator of the machine where the program is installed. Instructions on installing and running meme are provided here:
http://meme.sdsc.edu/meme/meme-install.html

The MEME executable should either be specified in the system PATH variable, user's PATH variable, or the absolute path should be listed in the Configure.ini file.

MultiFinder also requires a number of support programs designed to work with AlignACE-formatted motif files. These programs have been developed previously by another group [1] and are listed below (the download location of each program is listed).

ScanACE [1,2]
The ScanACE distribution currently available has become incompatible with newer compilers. A version that will compile is available on the MultiFinder download site and is called scanace-code.tar. The program can be unpacked and compiled as follows:

```
tar –xvf scanace-code.tar
```

This will create a directory containing the source code to ScanACE. Change directories into the scanace-code directory created by the tar function. Now copy the make-ScanACE file to makefile and then compile with the make function:

```
cp make-ScanACE makefile
make
```

This will generate an executable version of ScanACE that can either be placed in the user's PATH variable or left in place. If the file is left in place, the absolute path to ScanACE must be specified in the Configure.ini file. The original ScanACE file can be found at:
http://atlas.med.harvard.edu/download/scanace_2003.tar.gz

However, some compilers will have difficulty with the syntax of the original ScanACE, so *for reviewers' purposes we recommend using the modified version available in the MultiFinder download directory.*

CompareACE [1,2]
http://atlas.med.harvard.edu/download/compareace.tar.gz
The CompareACE distribution comes precompiled. Once the file is downloaded the program must be unpacked using the following command:

```
gunzip compareace.tar.gz
tar -xvf compareace.tar
```

These commands will generate a compiled version of CompareACE. The CompareACE program must either be defined in the users PATH variable or the absolute path must be specified in the Configure.ini file.

MotifStats [1]
The MotifStats distribution currently available has become incompatible with newer compilers. *For reviewers' purposes, a version that will compile (with warnings) is available on the Multifinder download site and is called motifstats_2003.tar.* The program can be uncompressed and compiled using the following steps:

```
tar –xvf scanace motifstats_2003.tar
```

This will create a directory containing the source code to MotifStats. Change directories into the motifstats-code directory created by the tar function. Now copy the make-MotifStats file to makefile and then compile with the make function:

```
cp make-MotifStats makefile
make
```

This will generate an executable version of MotifStats that can either be placed in the users PATH variable or left in place. If the file is left in place, the absolute path to MotifStats must be specified in the Configure.ini file. The unmodified version of MotifStats can be found at this address:
http://atlas.med.harvard.edu/download/motifstats_2003.tar.gz

Tree [1] and **list_clusters.pl** [1]
http://atlas.med.harvard.edu/download/clustering.tar.gz
These programs come as a zipped tarred file that must be uncompressed before using:

```
tar –xvf clustering.tar
```

The resulting programs must be put into a directory specified in the users PATH variable or the absolute path must be specified in the Configure.ini file.

Once the motif finders and the supporting programs have been installed, the configuration script (Configure.sh) can be run. This script will search for the absolute path to each of the programs listed above. If these programs have been installed and the PATH variable has been set for each program, the configure script will find the absolute path. If the PATH has not been defined one or more program, the output from the Configure.sh script (Configure.ini) will have to edited to list the correct location of each program. After running Configure.sh, enter:

```
bash Configure.sh
```

Check the output of Configure.ini. The output should look something like this:

```
AlignACE:/usr/local/bin/AlignACE
BioProspector:/usr/local/bin/BioProspector
MDscan:/usr/local/bin/MDscan
meme:/usr/local/bin/meme
ScanACE:/usr/local/bin/ScanACE
```

```
CompareACE:/usr/local/bin/CompareACE
MotifStats:/usr/local/bin/MotifStats
Tree:/usr/local/bin/Tree
list_clusters.pl:/usr/local/bin/list_clusters.pl
```

If any of these paths are missing or incorrect in the Configure.ini script, then insert or edit a line with the correct absolute path to the specified program.

IV. Required supporting files for SequenceExtractor

SequenceExtractor requires sequence, alignment information, and genomic annotation to generate the support files required by MultiFinder. The human genome sequence should be obtained from UCSC genome site:
http://hgdownload.cse.ucsc.edu/goldenPath/hg17/chromosomes/
For convenience, we have made available in the MultiFinder download directory on our lab website, a compressed version of the human genome sequence (Hg17Dev.tar.gz).

The sequence should be FASTA-formatted sequence with one file per chromosome. The names of the files should use the format "chr<chromsome>.fa" – for example, chromosome 1 would be named "chr1.fa". Alignment files can also be obtained from the UCSC website. These must be in the axt file format described on the UCSC site. Each .axt file must be named using the convention "chr<chromosome>.axt" – for example, chromosome 1 would be named "chr1.axt" and chromosome X would be name "chrX.axt". Annotation listing gene positions is also available at the UCSC website.

The SequenceExtractor script uses RefSeq accession numbers. A list of accession numbers with the locations of the genes are provided in the refGene.txt file available at the UCSC website. The UCSC site contains many alignments between human and other organisms. Any alignment can be used by SequenceExtractor provided that the FASTA-formatted sequence files are from the same assembly of the genome. For example, you could use the alignments and sequence from Hg17. However, you cannot use the alignments from Hg16 and sequence from Hg17. A list of nonredundant genes is necessary for SequenceExtractor to generate a background without multiple copies of the same sequences. This can be obtained from Unigene. This file can also be in the form of a simple list provided by the user, if an alternate background is desired. The purpose of this file is to provide SequenceExtractor with the set of nonredundant genes to be used to generate the background. A suitable file is available from UniGene at:
ftp://ftp.ncbi.nih.gov/repository/UniGene/Homo_sapiens/ Hs.seq.uniq.gz

This file should be unzipped using a tool like gunzip and then renamed "UniqueUnigene.txt". The purpose of this file is to designate a group of nonredundant genes for the background set during motif finding. SequenceExtractor.pl will create background files for the different motif finders using this set of genes. The user can choose to use a different set of genes for the background. A file containing a simple list of RefSeq accession numbers can be used in place UniqueUnigene.txt file. All of these files should be put into the same directory. When

SequenceExtractor is called, it will require these files to generate the sequence files for motif finding, and the directory where these files are located must be specified:

```
perl SequenceExtractor.pl -i ./SequenceExtractorTest/SequenceExtractorTest.txt -o
./SequenceExtractorTest -b /data/In/Human/Hg17Dev -m rc -s 5 -r u -d 1000 -l 20 -f 5
```

SequenceExtractor will use this annotation and sequence information to extract sequence from the genome according to the user's specifications. An example set of annotation and sequence files has been provided in order to allow a user to test SequenceExtractor. This files are located on the MultiFinder website in a compressed directory called Hg17Dev.tar.gz/. These files can be unpacked using the following commands:

```
gunzip Hg17Dev.tar.gz
tar –xvf Hg17Dev.tar
```

V. Using MultiFinder, SequenceExtractor, and BlockFilter:

Multifinder.pl, SequenceExtractor.pl and BlockFilter.pl are all Perl scripts. These three scripts require that the supporting Perl modules supplied with MultiFinder are installed either in the same directory as the Perl scripts or in the Perl module library.

1. Unzip and untar the archive. This step creates the required directory structure.
2. If needed, install Perl. ActivePerl is available at: http://www.activestate.com/Products/ActivePerl
3. MultiFinder generates graphics for each motif. These graphics are SVG formatted and are generated using the XML::Writer library available from CPAN. If you get your Perl distribution from Active State the XML::Writer library can be installed using the Perl packet manager PPM that comes with that distribution. Please go to the ActiveState web site for further instructions on how to download and install Perl. Perl is generally already installed on most linux machines. However, if it is not installed, the administrator of the machine should do the installation.
4. If the motif finders and supporting programs have already been installed but are not available at the command line, the paths to individual programs can be explicitly defined by changing the appropriate lines in the AnalysisFunctions.pm module. This will not be needed if the motif finders and supporting programs have been installed to be available at the command line.
5. On UNIX, be sure to have correct file and directory permissions.
6. Example input and output files have been provided to determine if the scripts are working properly.

Until publication of this manuscript, MultiFinder, SequenceExtractor, and BlockFilter will be freely available from either the *BMC Bioinformatics* website or at our lab website: http://the_brain.bwh.harvard.edu/GBMF/multifinder_dev_10Feb06/
using the reviewer login:

Username: GBreviewer
Password: 1htpgps

A compressed version is also available from the web site called multifinder_dev_10Feb06.tar.gz. This file can be unpacked using the following commands:

```
gunzip multifinder_dev_10Feb06.tar.gz
tar –xvf multifinder_dev_10Feb06.tar
```

This will create a directory with all of the required Perl scripts and modules as well as a number of test examples.

On that same website, we have also provided the following example input and output files and directories for SequenceExtractor, MultiFinder, and BlockFilter, within the following example input ("*Test") and example output ("*Example") directories:

SequenceExtractorExample
SequenceExtractorTest
BlockFilterExample
BlockFilterTest
MultiFinderExample
muscle_test (directory containing example input files for use with MultiFinder)

The necessary directories and files for use with the example command lines provided in this instructions file are copied when a user downloads the entire "multifinder_dev_20Jan06" directory.

Usage:

MultiFinder:

The Multifinder script can be called from the command line or from a shell script. If the MultiFinder script and the supporting modules have been installed and the PATH variable has been set, then the MultiFinder script can be called from any directory. If the script and supporting libraries have not been installed, they can be used from within the directory where they are located provided the modules are in the same directory as the MultiFinder script.

The following is the usage of MultiFinder:

```
perl MultiFinder.pl -i <file.seq> <options>
```

Options:

-i Input sequence file in FASTA format using a RefSeq accession number with an index for multiple sequences from the same region. For example:

```
>NM_012345_1
AGGTACCTGACCCAATACG.....
>NM_012345_2
AGGTACCTGACCCAATACG.....
>NM_782130_1
AGGTACCTGACCCAATACG.....
```

-o Output directory

-b Background sequence file in FASTA format using the same format as the input sequence file. This file should contain all of the sequences from input set as well as the additional sequences from the background.

-p Markov model of the background. This is a word frequency representation of the background.

-s Sequence file used with ScanACE. This sequence file contains all of the sequences from the background with spacing sequences to maintain the register of the sequence with respect to the start site of the given gene. This file is used to calculate positional specificity scores.

-w Minimum width of motif search range (default 8). This is the minimum motif width to be searched. This value may not be less than 6.

-W Maximum width of motif search range (default 12). This is the maximum motif width to be searched. This value may not be larger than 18.

-m Number of motifs to find at each width (default 10). The motif finders will attempt to find this number of motifs at each specified motif width. A maximum of 30 motifs is permitted, based upon inherent limitations of MDscan.

-c Correlation coefficient threshold used to eliminate duplicates (default 0.6). This is the Pearson correlation coefficient to be used by CompareACE.

-h Maximum number of ScanACE hits (default 10000)

-t Target scoring functions used to cluster motifs (default gsmbr). This is the scoring function(s) used:
g = group specificity
s = site specificity
m = motif finder-specific motif score
b = bit score

-d Distance used for positional specificity calculations (default 1000). This is the distance that MotifStats considers when calculating positional specificity

-l List of known motifs in AlignACE format. This file should also contain the names of the transcription factor binding site motifs.

-f MotifFinding programs to use (default abmd)
a AlignACE
d MDscan
b BioProspector
m MEME

Example input and output files have been provided with the MultiFinder distribution to allow users to test their installation of our programs. Once the proper supporting programs have been installed, the following command can be given within the directory where MultiFinder and its supporting modules are located. The following example command will call MultiFinder to search for 10 motifs of width 8-10 for each of the four motif finders. In this example, the motif search will be limited to the 1 kb regions upstream of genes differentially expressed in human skeletal muscle. After running this command (should take less than one hour to finish), the user can check the output files created in directory ./MultiFinderTest/ against the example output files we have provided in directory ./MultiFinderExample/. However, please note that the results will not be identical because some of the motif finders are stochastic and so will give different results from run to run. However, the results should be in the same general format and should be similar overall.

The following is the example described above (the following is all one long command line without line breaks – the directory names are simply long enough to have been wrapped around to the next line by Microsoft Word):

```
perl ./MultiFinder.pl -i
./SequenceExtractorExample/SequenceExtractorExampleMaskedConserved1000bpUpstream.seq
-o ./MultiFinderTest -b
./SequenceExtractorExample/BackgroundMaskedConserved1000bpUpstream.seq -p
./SequenceExtractorExample/BackgroundMaskedConserved1000bpUpstream.bfile -s
./SequenceExtractorExample/ScanMaskedConserved1000bpUpstream.seq -w 8 -W 10 -m 10 -c
0.6 -h 1000 -t gsmbr -d 1000 -l ./muscle_motifs.ace -f adbm
```

A shell script (filename “MultiFinderTest.sh”) is provided with the distribution. The path to the directory where the MultiFinder.pl script is located must be changed to reflect the directory structure of the user’s machine.

SequenceExtractor:

The SequenceExtractor script is used to generate the sequence and background files used by MultiFinder. SequenceExtractor uses genomic sequence, sequence annotation and sequence alignment files to generate input and background sets.

The following is the usage of SequenceExtractor:
perl SequenceExtractor.pl -i <file.seq> <options>

-i	Text file containing RefSeq accession numbers
-o	Output directory
-b	Directory containing background files
-m	Regions to mask r = repeats c = non-conserved regions as determined from *.axt files

-r	Regions to search
	u = upstream
	i = intronic
	d = downstream

-d	Number of bases to search upstream of the transcriptional start site
-l	Minimum acceptable sequence length

Once the directory containing SquenceExtractor and the supporting files is installed, the following example command can be used to test SequenceExtractor using the example input file we have provided:

```
perl SequenceExtractor.pl -i ./SequenceExtractorTest/SequenceExtractorTest.txt -o
./SequenceExtractorTest -b /data/In/Human/Hg17Dev -m rc -s 5 -r u -d 1000 -l 20 -f 5
```

The results from this search should be the same as those provided in the SequenceExtractorExample directory, with the exception of the randomly generated sequence files which would be different. The files generated can be used directly by MultiFinder.

A shell script (filename “SequenceExtractorTest.sh”) has been provided with the distribution that will run SequenceExtractor.pl. The directories in this script must be modified to represent the user’s directory structure.

BlockFilter:

The functionality of the BlockFilter script is already contained within MultiFinder. However, since a user may find the filtering criteria used by BlockFilter useful in other applications, we are also providing BlockFilter as a separate program. This script is designed to search for contiguous regions of sequence preference within motifs and eliminate motifs not containing the user-specified degree of sequence preference. BlockFilter looks for two types of motifs: those containing a single block of sequence preference, and those containing blocks of sequence preference separated by regions without significant sequence preference.

The following is the usage of BlockFilter:

```
perl BlockFilter.pl –i <file.ace> <options>
```

Options

-i Input file in AlignACE format

-o Output file in AlignACE format

-d Number of contiguous bases in a motif over the threshold information content required to define a single block motif

-p Number of contiguous bases in a motif over the threshold information content required to define a multiple block motif. Two or more contiguous blocks over this length must be found to define a multiple block motif.

-c Correlation coefficient threshold used to determine if a multi-block motif is palindromic or contains a direct repeat

-s Minimum bits of information content for a motif to be called as part of a block. (default 0.5)

The following is an example using the example input file we have provided:

```
perl ./BlockFilter.pl -i
./MultiFinderExample/SequenceExtractorExampleMaskedConserved1000bpUpstreamCombined
GroupAll.ace -o
./BlockFilterExample/SequenceExtractorExampleMaskedConserved1000bpUpstreamCombined
GroupAllFiltered.ace -d 4 -p 3 -c 0.6 -s 0.5
```

This example will look for single-block motifs with at least 4 positions of contiguous sequence preference and for multi-block motifs containing at least 2 blocks each with at least 3 positions of contiguous sequence preference. Sequence preference is defined as nucleotide positions with at least 0.5 bits of information. The correlation cutoff for identifying palindromes and direct repeats is 0.6. A shell script (filename "BlockFilterTest.sh") has been provided to test this function. The directories do not need to be modified for this test, but they need to be modified for the user's input motif sets.

VI. References:

1. Hughes JD, Estep PW, Tavazoie S, Church GM: **Computational identification of cis-regulatory elements associated with groups of functionally related genes in *Saccharomyces cerevisiae***. *J Mol Biol* 2000, **296**:1205-1214.
2. Roth FP, Hughes JD, Estep PW, Church GM: **Finding DNA regulatory motifs within unaligned noncoding sequences clustered by whole-genome mRNA quantitation**. *Nat Biotechnol* 1998, **16**:939-945.
3. Liu X, Brutlag D, Liu J: **BioProspector: discovering conserved DNA motifs in upstream regulatory regions of co-expressed genes**. In *Pac Symp Biocomput*: 2001:127-138.
4. Liu X, Brutlag D, Liu J: **An algorithm for finding protein–DNA binding sites with applications to chromatin-immunoprecipitation microarray experiments**. *Nat Biotechnol* 2002, **20**:835-839.
5. Bailey T, Elkan C: **The value of prior knowledge in discovering motifs with MEME**. In *Proc Int Conf Intell Syst Mol Biol*: 1995:21-29.