FADE: Fast Atomic Density Evaluator

What is FADE?

The Fast Atomic Density Evaluator (FADE) is a program for computing protein shape descriptors and for analyzing shape complementarity at protein-protein docking interfaces [1]. The "atomic density" descriptors computed by FADE follow the methodology outlined in [2]. The computational methods employed by FADE use the Fast Fourier Transform (FFT) to measure atomic density at a rate that is orders of magnitude faster than direct methods. In addition, techniques for directly measuring shape complementarity at docking interfaces have been implemented.

Why use FADE?

The "topographical" information produced by FADE can help researchers easily pinpoint the most prominent features of a protein, regions which are likely to participate in interactions with other molecules. In addition to providing shape descriptors to aid in analyzing single molecules, FADE can directly evaluate the level of shape complementarity for docked protein-protein complexes.

FADE is extremely easy to use. The program can compute shape descriptors or evaluate shape complementarity given only the PDB file(s) for the molecule(s). The program automatically sets the grid sizes needed to run FFT's for optimal efficiency, without requiring user input.

Who and Where

FADE was developed at the San Diego Supercomputer Center by Julie C. Mitchell. The Fast Fourier Transform and convolution routines were adopted from the DOT program using code written by Lynn F. Ten Eyck and Jeff G. Mandell. FADE is distributed by the Computational Center for Macromolecular Structures. Please visit our website http://www.sdsc.edu/CCMS

References

Please reference the following manuscripts in any scientific publications that use FADE's atomic density methods for shape characterization:

[1] Mitchell, J.C., Kerr, R. and Ten Eyck, L.F. (2001) "Rapid Atomic Density Methods for Molecular Shape Characterization," J. Mol. Graph. Model, 19(3), pp. 324-329.

[2] Kuhn, L.A., Siani, M.A., Pique, M.E., Fisher, C.L., Getzoff, E.D., and Tainer, J.A. (1992) "The interdependence of protein surface topography and bound water molecules revealed by surface accessibility and fractal density measures," J. Mol. Biol., 228, pp. 13-22.

Installing FADE

Supported Platforms

FADE has been compiled and successfully run on all of the following platforms:

Platform	OS Version
SunOS (solaris)	5.5
IRIX	6.5
IRIX64	6.2
OSF1 (alpha)	4.0
Linux	2.2.16
Darwin (Mac OS X)	10beta

To compile FADE on Unix-based systems, C and Fortran (optional) compilers are needed. Compiled executables are available for common Unix platforms and Macintosh OS 8/9/10. The Mac OS 8/9 version is available by special request to mitchell@sdsc.edu

Installing FADE

Installing FADE should be very simple.

1. Download the file "fadepadre.tar.gz"
2. Type "gunzip fadepadre.tar.gz"
3. Type "tar xf fadepadre.tar"
4. Type "cd FADEPADRE"
5. Type "make" (Note: you may skip this if you use a precompiled binary)

Hopefully, this will work.

Some caveats

If you do not have a Fortran compiler, you may copy the file "fft.c" from the "src/nof77" directory into the "src" directory and recompile. However, since the Fortran code will run faster than the C code, it is better to use one of our precompiled binaries in this case.

If you do not have "gmake" installed on your system, change the value of "MAKE" in the root "Makefile" to "make" For some systems, you may also need to change the C compiler. You can do this permanently by editing the "configure" file in the "conf" directory. You can do this temporarily (after configuration) by editing the "Makefile" in the "src" directory. If you are still having problems, please use the compiled executable for your system.

Bug reports

Please report any bugs to mitchell@sdsc.edu. Include the terminal output for the example you are having problems with, and indicate the operating system you are using.

Running FADE

FADE recognizes PDB files and files of the format "x y z …" If a ".pdb/.ent/.pqr/.qri" extension is not found, the xyz format is assumed.

Defaults

Usage: FADE [-fgmnopqr] m1.pdb [r0 r1] [d0 d1] [[m2.pdb] [R0 R1] [D0 D1]]

Distance and density exponent bounds

```
r0 = 0          r1 = 3          d0  = 0.0   d1  = 5.0
R0 = 0          R1 = 3          D0  = 0.0   D1  = 5.0
```

Problem sizes

```
NumConv = 10          ResLev  = 1
```

Output file types

```
outList = 1       outPDB  = 0       outGrid = 0
```

Compute density exponents for a single molecule

To compute density exponents for a single molecule, with **defaults**:

```
FADE    mol.pdb
```

To compute density exponents for a single molecule with **manual distance bounds**:

```
FADE    mol.pdb  r0  r1
```

To compute density exponents for a single molecule with **manual distance and exponent bounds**:

```
FADE    mol.pdb  r0  r1  d0  d1
```

Compute shape complementarity scores for a pair of molecules

To compute shape complementarity scores for two molecules, with **defaults**:

```
FADE  mol_1.pdb  mol_2.pdb
```

To compute shape complementarity scores for two molecules, with **defaults for the second molecule**:

```
FADE    mol_1.pdb  r0  r1  d0  d1     mol_2.pdb
```

and so on …

Flags recognized by FADE

-f Write “FULL” .fad output file containing points, distances, exponents and all values of the radial counting function. This option is only available for single molecules.

-g Write “GRID” .grd output. This option is only available for single molecules. Grid points will be written one per line. The fastest-cycling exponent is “x”, then “y”, then “z.”

-m= Median density exponent, ex0 (real). This sets the median density exponent used for scoring docked complexes and for creating temperature factor entries in PDB output files. The default value is 2.8, which is equivalent to “-m=2.8”

-n= Number of convolutions (integer). This sets the the maximum radius for which the radial counting function will be computed. To save time, FADE does not do convolutions for radii which are less than (r0-1), so the total number of convolutions actually performed may turn out to be less than this number. The default number of convolutions is 10, and we do not recommend significantly increasing or decreasing this number.

-o= Override automatic output naming. It changes the base file name for output to “Myname” if given the flag “-o=Myname”

-p Write “PDB” output. Density exponents or complementarity scores are used to construct temperature factor entries in PDB output files.

-q Quiet mode. Great for scripting.

-r= Resolution level (integer). This sets the grid spacing to 1/L if given the flag “-r=L.” The default resolution level is 1.

Note: FADE recognizes both lower and upper case versions of all its flags.

Setting the parameters used by FADE

Distance bounds

The distance bounds determine the range of distances allowed between output points and the molecule(s). For a single molecule computation, setting r0=1 and r1=3 will return points which are approximately 1-3 Angstroms from an atomic center.

For a docking interface, setting r0=1, r1=3, R0=2, and R1=3 would return all grid points which are approximately 1-3 Angstroms from an atomic center in the first molecule AND which are approximately 2-3 Angstroms from an atomic center in the second molecule. The default values set r0=R0=0 and r1=R1=3. If one is examining structures in which there is a collision (ie: some conformational changes are needed) it may be more advisable to set r0=R0=1 so that the colliding regions do not contribute to the complementarity score.

Density exponent bounds

The density exponent bounds determine the range of exponents allowed for output points. For a single molecule computation, smaller values (< 2.5) typically indicate crevices or grooves and larger values (> 3.0) typically indicate protrusions. Thus, to return only points which lie in crevices and grooves, one might set d0=0.0 and d1=2.5.

For a docking interfaces, setting d0=0.0, d1=2.5, D0=3.0 and D1=5.0 would return all points having exponent between 0.0 and 2.5 with respect to the first molecule and exponent between 2.5 and 3.0 with respect to the second molecule. Typically, such points would suggest a place where a protrusion in the second molecule has been docked into a crevice on the first molecule. By setting setting d0=0.0, d1=2.5, D0=0.0 and D1=2.5, one can look for crevice-crevice regions in the docking interface.

Number of convolutions

This option is set using the -n flag. The number of convolutions sets the maximum radius for which the radial counting function will be calculated. We don't really recommend changing this from the default of 10 unless you have reason to do so.

Median exponent

This option is set using the "-m" flag. The median exponent determines the exponent value for which local geometry is expected to be flat. The default is set to 2.8. A somewhat smaller value may be best for larger molecules and conversely. We are looking into (possibly system specific) ways of setting this parameter optimally. Fortunately, while changing the value of the parameter changes the values of complementarity scores somewhat, the relative rank of solutions tends to be fairly stable under small perturbations of the parameter.

FADE Output

LIST (.fad,.dad)

FADE will always produce a **list of points, distances and scores**. For a single molecule, the output FADE atomic density (.fad) format is

```
x  y  z  distance  exponent
```

Here, "x", "y" and "z" give the coordinates of a grid. The value of "sscore" gives the shape score, which results by subtracting the median exponent from the computed exponent.

For a pair molecules, the output docking atomic density (.dad) format is

```
x  y  z  dist1  ex1   dist2  ex2  cscore
```

Here, cscore is the shape complementarity score at (x,y,z), which depends on the median exponent ex0.

FULL LIST (.fad.all)

FADE will produce a **list of points, counting function values, distances and scores**. This option is only available for single molecule computations using the "-f" flag. The format is the same as for the list output option, except that the counting function values for r=(1,…,numConv) are appended to each line.

PDB (.fad.pdb)

To aid in visualizing results, FADE will produce a **PDB file** with one ATOM record for each returned grid point. The temperature factors are computed using the formulas

$$T = 50 + 25\ (ex - ex0) \qquad \text{(single molecule)}$$
$$T = 50 - 25\ (ex1 - ex0)\ (ex2 - ex0) \qquad \text{(two molecules)}$$

To view the results of a FADE output PDB file, you can open it with Rasmol and type:

```
spacefill
color temperature
```

Red (high temperature) will denote high density regions in crevices for single molecules. For docked complexes, red will indicate regions of high shape complementarity (which have negative shape complementarity scores.)

Note: Temperature factors in the PDB output files are bounded by 1 and 99. Do not use them to reconstruct shape and complementarity scores.

GRID (.fad.grd)

For a single molecule computation, FADE can produce a three-dimensional **grid of results** for the computed exponents using the "-g" flag. The the output grid density (.fad.grd) format is one entry per line, with "x" being the fastest-cycling exponent, followed by "y" and "z." The grid center and grid sizes are displayed in the header at the top of the file.

SCORE (.sad)

The scored atomic density. (.sad) file records the number of points returned in the .fad file, along with the total and average scores. The complementarity scores are used for a docking interface, and for a single molecule, the density exponents are used.

Supplied Utilities

zeroT

zeroT can be useful when viewing the FADE results stored in PDB files. Running zeroT on the original PDB file for the molecule(s) will set their temperature factors to zero. When the experimental temperature factors are displayed along with the FADE results, the temperature scale can sometimes be skewed in undesirable ways. To best highlight regions of high and low complementarity, you should run zeroT on PDB files before displaying them with FADE's output files. Syntax is:

```
zeroT  oldfile.pdb  newfile.pdb
```

getchain

getchain extracts chains with a given identifier from a pdb file. This can be useful when doing docking calculations in which one wishes to run FADE on chains A and B against chains C and D of a particular PDB structure. In this case, you could create the necessary PDB files using the two commands

```
getchain  mol.pdb  mol_AB.pdb  AB
getchain  mol.pdb  mol_CD.pdb  CD
```

The underline "_" symbol indicates that getchain should retrieve entries without a chain identifier (water, for example).

pdbshift

pdbshift can rotate and translate a PDB file using six rigid body coordinates. The first three coordinates give an offset to be taken after the molecule is rotated. The last three coordinates give Euler angles specifying a rotation matrix. To view or change the Euler angle convention used, have a look at the "dotrots.c" file in the "src" directory. The syntax is

```
pdbshift  mol.pdb  mol_trans.pdb  x  y  z  phi  theta  psi
```

to translate and rotate

```
pdbshift  mol.pdb  mol_trans.pdb  x  y  z
```

to translate or

```
pdbshift  mol.pdb  mol_cen.pdb
```

to center the PDB structure.

fadebatch

A utility is provided to run FADE for a list of docking configurations defined in terms of rigid body transformations, such as that output by the docking program DOT. The syntax is

```
fadebatch  moving.pdb  still.pdb  basename  configs.list
```

where basename gives the output base name for transformations of the moving molecule and configs.list gives rigid body coordinates for the transformations. Input file format for configs.list is "energy x y z phi theta psi"

Interpreting FADE Output

The density exponents returned by FADE reflect the rate of increase in atomic neighbors about points near the molecular surface. A uniform distribution of atoms in 3D space should return a density exponent of approximately 3.0. That is, the number of neighboring atoms contained in a ball of radius r increases roughly as r^3. Since molecules are not perfectly dense, one typically sees a median density exponent in the range 2.8-2.9. Values higher than the median exponent suggest a crevice, while lower values suggest a protrusion. To compute shape complementarity, a quadratic formula returns a negative number in the case of complementarity and a positive number in the case of a mismatch.

Troubleshooting and Known Bugs

FADE and multiple structures

Q: "Why does my output look funny? FADE thinks everything is a crevice."

A: You probably have multiple structures in your PDB file, which is frequently the case with structures obtained by NMR. Because FADE's methods rely on an atom counting scheme, this could skew the results.

Shell script wrappers

Q: "I am having problems running the scripts in the /bin/work directory."

A: The files in this directory are wrappers to call executables in the /bin/exec and /bin/utils directories. The scripts automatically figure out which system you are running and call the appropriate executable. Your problem likely comes from the fact that you do not have the Korn shell (ksh) installed on your Unix system. This shell is used to allow more than 9 arguments to be called in the script. If you do not have the Korn shell, you can try executing the script using other shells (for example, try "sh FADE …" or "bash FADE …" to see if this fixes the problem. Alternately, you can just use the actual executables located in the /bin/exec and /bin/utils directories.

Mac OS X

Q: "I double-click on the FADE executable, but it doesn't open."

A: Executables for Macintosh OS X are compiled as native Unix tools. This means that you need to run them from a Unix terminal. There is a program called "Terminal" included with Mac OS X for this purpose. If you are not familiar with navigating Unix-based filesystems, the book "Unix in a Nutshell" by Arnold Robbins can help you get started.