DomainFinder User's Manual

Konrad Hinsen, hinsen@cnrs-orleans.fr

v1.1, 2001-9-5

DomainFinder is an interactive program for the determination and characterization of dynamical domains in proteins. Its key features are computational efficiency, ease of use, and export of results for visualization and further analysis.

Contents

1 Introduction

DomainFinder is an interactive program for the determination and characterization of dynamical domains in proteins. A *dynamical domain* is a region in a protein which can move essentially like a rigid body relative to other regions. Many, but not all, proteins have dynamical domains, and if they do, the relative movements of the domains are usually related to the function of the protein. The identification of dynamical domains is therefore useful in understanding the function of the protein. However, there are other situations in which the knowledge of dynamical domains is helpful. In structure determination, it can help to predict whether complexation with a ligand, crystal packing, or other external influences can lead to important conformational changes. In protein engineering, it can indicate whether a given modification is likely to change the dynamical behavior of the protein. In experimental observations of protein motion, it can suggest regions of particular interest. In numerical simulations, it can point out the slow motions whose correct sampling must be verified.

DomainFinder is based on new theoretical methods which permit the identification of dynamical domains from a single conformation with almost negligible computational effort. Any state-of-the-art desktop computer should be sufficient for analyzing proteins of about 1000 amino acid residues in a few minutes, and even the largest known protein structures can be analyzed in less than an hour using common workstations or high-end PCs. DomainFinder can also determine dynamical domains from a comparison of two given conformations of the same protein.

DomainFinder is written in *Python*, a high-level object programming language that is particularly well suited to the demands of scientific computations. The speed-critical parts are implemented in C. For common operations it makes use of the *Molecular Modeling Toolkit*, a library of Python code for molecular modeling and simulation applications. The results of a domain analysis can be saved with all details in an MMTK data file, which permits all kinds of further analysis.

2 Theoretical basis

A detailed description of the theoretical methods used in DomainFinder can be found in Ref. 1 (1). This section provides a brief summary that everyone should read before using DomainFinder.

Since dynamical domains are by definition large regions in a protein, DomainFinder does not look at all atoms, but only at the C-alpha atoms, which define the backbone. Specifically, DomainFinder looks at *changes* in the relative positions of C-alpha atoms; these changes describe the deformation of the protein while it undergoes a specific motion. To characterize the deformation of each part of the protein, DomainFinder calculates a deformation energy for each atom, which depends on the changes in the distance of this atom from each of its close neighbors. A low deformation energy indicates relatively rigid regions, which are candidates for dynamical domains, whereas high deformation energies indicate flexible regions. Dynamical domains are identified by grouping the rigid regions according to their overall motion; a dynamical domain is thus by definition a sufficiently rigid region whose parts move in a sufficiently similar way. This procedure requires two input parameters that the user must provide: a deformation threshold, up to which a region is considered sufficiently rigid, and a similarity threshold, up to which the motion of subregions is considered sufficiently similar.

In order to analyze protein motions, DomainFinder needs some input data describing them. One possibility is the use of two different conformations; the difference between these conformations describes one possible motion. Typically these conformations are obtained experimentally (by crystallography or NMR), and differ by the presence or absence of certain ligands. This mode of operation amounts to an analysis of experimental data.

However, there are only few proteins for which two substantially different conformations are known experimentally. DomainFinder can also use calculational methods to obtain the necessary information about domain motions, using only a single conformation as input. The technique used by DomainFinder is a variant of *normal mode analysis* which has been optimized for the purpose of domain motion study. It is described in detail in Ref. 2 (2). Even if multiple experimental structures are available, the normal mode approach is in general preferable, because it yields clearer domain delimitations and more detail. This is not surprising, since two conformations necessarily contain only a small subset of the full protein dynamics.

3 Preparing your protein structures

For a normal-mode-based domain analysis, only a single protein structure is required, which must be available in the Protein Data Bank (PDB) format. DomainFinder makes an effort to deal with all common variants of the PDB format, but a correct interpretation is guaranteed only for files that follow the PDB format definition exactly, for example files from the Protein Data Bank itself. The PDB file may contain non-protein molecules (DNA strands, small molecules, etc.), but DomainFinder will extract only the peptide chains.

For a comparison of two conformations, you must have two compatible conformations, i.e. two PDB files that contain the same residues of a protein. DomainFinder verifies that the two files contain the same number of chains, and for each chain the same number of residues. However, it does not insist that the residue types be equal (in order to allow a comparison of mutations), and it cannot verify that corresponding residues in the files actually describe the same residues in reality. Since many published protein structures are incomplete, i.e. some residues are missing, you must take care that the two protein conformations you have are compatible, if necessary by manually editing the PDB files to remove residues that are in one conformation but not in the other. DomainFinder provides a visual help to verify that your conformations are indeed compatible.

4 Normal mode based analysis

After starting DomainFinder, load your protein structure using “Load reference structure...” in the File menu. DomainFinder will then propose reasonable values for the number of calculated modes and the number of modes kept for analysis. However, you may modify these numbers if you wish. The number of calculated modes determines the accuracy of the normal mode calculations; the more modes you calculate, the better the description of the motions. The value proposed by DomainFinder is a rather small, but reasonable, value; you might want to increase it to obtain better results. In contrast, the number of modes kept for analysis has no influence on the quality of the results. It exists purely for efficiency reasons: keeping all calculated modes would slow down the analysis and waste disk space if the modes are saved in a file. There is little reason to increase the proposed number, because the additional modes typically do not describe domain motions are are therefore not useful for the domain analysis.

To start the normal mode calculation, press the button labeled “Calculate modes”. Depending on the size of the protein and the speed of your computer, the calculation can take from a few seconds to a few hours. For example, the calculation of 100 modes for a 500-residue protein takes 90 seconds on a 90 MHz Pentium PC; the same calculation takes 22 seconds on an IBM RS/6000 Model 43P-140. When the calculation is finished, DomainFinder displays a list of the modes with non-zero frequencies (the first number is seven, because there are six zero-frequency modes) and the actual number of calculated modes. This number is in general different from the one you entered, because not all values are possible for technical reasons (see the discussion on Fourier bases in Ref. 2 (2)). If the number of modes you ask for is equal to or larger than the number of residues in the protein, DomainFinder will calculate all possible modes (three times the number of residues), because calculating fewer modes would not be more efficient.

You must now select the modes that you want to use for the deformation and domain analysis. To help you with this choice, the normal mode list indicates the average deformation energy per residue for each mode. As explained in 2 (section 2), a deformation energy is associated with every atom; low values characterize rigid regions, whereas high values indicate flexible regions. A low average deformation energy thus indicates a mode with large rigid regions, which has a good chance of describing domain motions. There is no simple recipe for selecting an optimal set of modes (otherwise DomainFinder would apply it automatically!). As a general guideline, look for jumps in the average deformation energy from one mode to the next, and choose all modes before such a jump; there is no justification for selecting only some out of a set of modes with very similar deformation energies. Start with few modes, and add more modes only if you are not satisfied with the amount of detail in the domain analysis.

After selecting the modes, you must choose the deformation threshold that defines which regions are sufficiently rigid to be candidates for domains. This choice is related to the mode selection; to have a reasonable number of rigid regions, the deformation threshold should be of the same order of magnitude as the average deformation energy of the modes you have chosen. A higher deformation threshold leads to larger rigid regions; if you find later that your domains cover too small a part of the protein, you should increase the deformation threshold.

The following steps are the same as for an analysis based on comparing two conformations, and are described in sections 6 (6) and 7 (7).

If you wish to stop working on an analysis and continue it later, you can save time by saving the already calculated normal modes to a file and then later loading this file instead of the input structure. This is done using the File menu entries “Save modes...” and “Load modes...”.

5 Conformation based analysis

After starting DomainFinder, click on the button labeled “Conformation comparison” to switch to conformation mode. You can switch between modes at any time if you wish to use both modes on the same protein. Load the reference structure via the File menu entry “Load reference conformation”, then the second structure via “Load comparison conformation”. Note that it does make a difference which of the two conformations is used as a reference; better results are obtained if the reference conformation is the one with a larger exposed surface, i.e. an “open” conformation. Once the two conformations have been loaded, DomainFinder eliminates global translational and rotational motion between them, and displays the remaining RMS difference.

If you wish to verify that your two conformations are indeed compatible, you can select “Show both conformation” from the File menu. This will show both conformations together, one in black and one in red, with the two positions of each atom linked by a blue line. Erratic blue lines indicate a mismatch between the two configurations. This display can also serve as a first impression of the difference between the two conformations.

Protein structures obtained from experiment are always subject to experimental inaccuracies. These inaccuracies can distort the deformation energy calculation, since a random error on each atom position is indistinguishable from a high overall deformation. DomainFinder can eliminate such experimental noise while keeping the important large-scale differences between the conformations; the details of this procedure have been described in Ref. 1 (1). The amount of noise filtering can be adjusted using the slider labeled “Noise filter”. Its value is the amount by which the RMS distance between the conformations will be reduced by filtering, given in nm (nanometers). The default value of 0.01 nm is good for most cases; you should choose a larger value if one or both conformations have high experimental inaccuracies, as indicated e.g. by a high crystallographic resolution. A smaller value must be chosen if the total RMS distance between the conformations is very small, i.e. smaller than about 0.1 nm. You can verify the effect of the filter visually using the File menu entry “Show filter effect”. It produces a display showing the original comparison conformation and its filtered equivalent. The reference conformation is not changed by filtering. The File menu entry “RMS distances” shows the three RMS distances between reference, comparison, and filtered comparison conformations.

Next you must choose the deformation threshold, i.e. the deformation energy value that you consider the highest acceptable one for regions classified as rigid. The default value is a good starting point; you may have to increase it significantly for small proteins. Note that the deformation energy definition used by DomainFinder is larger than the one described in Ref. 1 (1) by a factor of five; this factor was introduced to make typical deformation energies for conformation-based analysis similar to those for normal mode-based analysis.

The following steps are the same as for an analysis based on normal modes, and are described in sections 6 (6) and 7 (7).

6 Deformation analysis

Much information about the slow motions of a protein can be obtained by looking at the deformation energies for each atom. The entry “Show deformation” in the Deformation menu shows the reference conformation with a color code representing the deformation energies. Blue atoms correspond to small deformation energies, green atoms are close to the deformation threshold, and red atoms are in particularly flexible regions of the protein. Blue and green atoms are thus below the deformation threshold and candidates for domains in a subsequent domain analysis; if you find that most of your atoms are yellow or red, you should increase the deformation threshold before starting the domain analysis. However, you should also consider the possibility that your protein has no domains, either because it is too flexible to allow a description of its slow motions by quasi-rigid substructures, or because the deformation associated with these motions is uniformly distributed over the whole protein.

Although the energy scale for the deformation energies is arbitrary (see Ref. 1 (1) for a detailed discussion), it is nevertheless an absolute scale independent of the specific protein. This means that deformation energy values can be compared between proteins and, in the case of a normal mode based analysis, between modes. If a certain protein requires a higher deformation threshold than another protein for a useful domain decomposition, this indicates that the first protein has a higher overall flexibility and less rigid domains. A domain decomposition should therefore never be published without quoting the deformation threshold used for it.

For a more detailed visualization, the deformation energy information can be exported in two formats. Using the Deformation menu entry “Write VRML file...”, you can write a VRML version of DomainFinder’s deformation visualization. With the menu entry “Write PDB file...” you can produce a PDB file which contains the deformation energy values coded in the temperature factor field for each atom. A temperature factor of 99 indicates atoms whose deformation energy exceeds twice the deformation threshold; values between zero and 79.2 correspond linearly to deformation energies between zero and twice the deformation threshold.

7 Domain analysis

The goal of a domain analysis is a decomposition of the protein into regions with distinct dynamical properties. The kinds of regions that can be distinguished depend on the domain decomposition approach that is used; in general, the outcome of two domain decompositions using different approaches will not only be different, but not even directly comparable. It is thus important to understand the capabilities and limitations of each approach.

The techniques implemented in DomainFinder allow an identification of three types of regions in a protein:

- Flexible regions, for which a description of the motion by rigid bodies is not useful. These regions are recognized during deformation analysis (see section 6 (6)) and not used at all during domain analysis.
- Rigid regions with uniform motion. These regions are recognized as rigid during deformation analysis; domain analysis then groups them together according to the similarity of their overall motion. However, these regions are *not* rigid bodies in any strict sense; they do show internal deformations, but these deformations do not destroy the uniformity of the overall motion. The term "dynamical domain" is best interpreted to describe these rigid regions only.
- Intermediate regions, whose internal deformation is sufficiently small everywhere, but systematic enough that over the size of the region it adds up to produce sufficiently different overall motion between extremal parts of the region. Such intermediate regions often occur in between dynamical domains.

Examples for all three types are shown in Ref. 1 (1).

The central parameter which allows a distinction between rigid and intermediate regions is the "domain coarseness" parameter. It specifies how similar the global motions in a region must be to be considered similar enough to form a domain (see Ref. 1 (1) for details). However, this parameter does not have one specific "best" value which one should find in order to obtain the "right" domain decomposition. It is a parameter which should be varied, and the ensemble of results for several values of this parameter provides the information for identifying domains and intermediate regions.

In order to obtain the motion parameters necessary for the domain analysis, DomainFinder first divides the protein into small cubic regions containing on average six residues. For each cube, six motion parameters are calculated, three for translation and three for rotation. In case of a normal mode based analysis, there are six parameters *per mode*; this is one reason why a normal mode based analysis usually gives better results. The cubes are then grouped into domains according to the similarity of their motion parameters. For a single value of the domain coarseness, it is not possible to distinguish between rigid and intermediate regions; the word "domain" therefore refers to both type of regions in the program.

After choosing a value for the domain coarseness, select "Show domains" from the Domains menu. This causes a window to be opened which shows the domain decomposition for this coarseness level. In the top left, the protein structure is drawn with various regions indicated by colors. The list to the right of the structure display contains all these regions with their color and size. The order in the list is significant; the best-defined domains are listed first, and the last item(s) frequently contain cubes that do not really belong to any recognizable domain. The bottom picture shows a parallel-axis plot of the motion parameters for all cubes, color-coded by domain. In this plot, each line represents one cube, and each vertical axis one motion parameter. For well-defined domains, the lines belonging to the same domain (i.e. same color) should be very close, whereas lines belonging to different domains should be clearly separated. A wide band of lines indicates an intermediate region. The plot provides both a verification of the domain decomposition and a first impression of the nature of the domains. However, it should be interpreted with caution; the eye tends to consider two lines with small differences in all axes more similar than two lines which coincide in some axes but differ significantly in others, although from a mathematical point of view both situations are equivalent.

It should be noted that the residues shown in black do not belong to any domain, for one of the following reasons:

- they are part of a cube with less than three points, which is too small to permit the calculation of the motion parameters
- they are in a cube whose average deformation energy is higher than the deformation threshold

The precision of the domain definitions is thus not one residue, but one cube. In practice this is of little importance, because dynamical domains are by definition big regions, certainly larger than a cube of six residues on average. However, this effect should be kept in mind, since it explains some features of the domain analysis that are at first surprising, e.g. the lack of a perfect symmetry in agreement with the symmetry of the molecule, or a slight dependence of the domains on the orientation of the input structures.

For more detailed information on a particular domain, click on that domain's entry in the domain list. This will open another window with information for this domain only. In the top left there is again the protein structure with just one domain highlighted. To its right, a list of all residues in the domain is shown. Below there is a parallel-axis plot showing only the cubes in this domain. Finally, there is an indication of the numerical similarity of the motion parameters within the domain. Two numerical similarity values are given, of which the first (larger) one is the similarity of the two most similar cubes, and the second (smaller) one is the similarity between this pair and the most different cube. The ratio between these two number is the domain coarseness which is necessary to consider the whole region as one domain. The small plot at the bottom shows one line per cube at the coarseness level required to keep that cube in the domain. You can use it to estimate the influence of a small change of coarseness on this domain: if there are many lines close to the current coarseness limit, the domain is likely to change significantly. Inversely, if the highest coarseness in the domain is clearly smaller than the current limit, a small variation will have no influence on the domain.

When you vary the domain coarseness limit, you will observe that some domains remain essentially the same, growing or shrinking only by small amounts and in response to significant coarseness variations, whereas others grow and shrink rapidly, or tend to break up into smaller parts as the coarseness limit is decreased. The first kind represents stable rigid regions, i.e. dynamical domains. The second kind represents intermediate regions. The parallel-axis plot at the bottom of the window helps in this classification by showing the variation of motion parameters within the domains at one glance.

Finally, DomainFinder lets you export the domain analysis results for visualization and further computational analysis by using the remaining entries in the Domains menu. “Write domain list...” writes a complete list of the domains and their residues to a text file. “Write PDB file...” creates a PDB file of the protein structure with the domain numbers coded in the “occupancy” field. A value of zero indicates a residue outside any domain, other values refer to the order of the domains in the domain list. “Write VRML file...” provides a VRML version of the color-coded structure in the domain window. “Save in MMTK format...” saves a Python dictionary in the object format used by the Molecular Modeling Toolkit; this dictionary contains entries for all variables of interest. The file can be loaded with the MMTK function `load()`.

8 User interface features

8.1 Files

All text files read or written by DomainFinder (e.g. PDB files, VRML files, and textual domain descriptions) can be compressed using the Unix utilities `compress` or `gzip`. It is sufficient to use filenames with the appropriate suffixes `.Z` or `.gz`; DomainFinder then automatically chooses the right format. Compression is especially useful for VRML files, which can become very large. Most VRML viewers can deal with compressed files directly, just like DomainFinder.

8.2 Protein structure visualization

Many windows in DomainFinder show a three-dimensional protein backbone structure. The representations can be moved, rotated, and scaled with the mouse. Press the left button for translation, the middle button for rotation, and the right button (with vertical movement) for scaling.

The structure visualization in DomainFinder is currently rather slow, which is especially visible during interactive manipulation. Future versions of DomainFinder will probably use a faster structure display technique.

9 Literature

1. **"Analysis of domain motions in large proteins" by K. Hinsen, A. Thomas, and M.J. Field (Proteins 34, 369-382, 1999)**

   This article describes the domain analysis methods implemented in DomainFinder in detail and discusses several applications.

2. **"Analysis of domain motions by approximate normal mode calculations" by K. Hinsen (Proteins, Proteins 33, 417-429, 1998)**

   This article describes the normal mode calculation technique that allows DomainFinder to treat even large proteins rapidly.