

Requirements for Tornado daughter application #1

version 1.1 by Christopher Bruns

August, 2005

1	Purpose.....	2
2	Introduction.....	2
3	Issues to be resolved	2
3.1	How much does this project overlap or extend the morph server project by Gerstein's group at Yale?.....	2
3.2	Can we collaborate with Mark Gerstein, Leslie Kuhn, Michael Thorpe, and/or Woolf to accumulate these tools?	3
4	Users	3
5	Functionality	3
5.1	User Interface:.....	3
5.2	Input:	3
5.3	Alignment	3
5.4	Interpolation.....	3
5.4.1	Global superposition	4
5.4.2	Multiple rigid body transformations	4
5.4.3	Torsion angle interpolation.....	4
5.4.4	Cartesian interpolation.....	4
5.5	Measure energy	5
5.6	Relax	5
5.7	Output	5
5.7.1	Coordinate files.....	5
5.7.2	Parameter trajectories.....	5
5.7.3	Movie files	5
5.7.4	Movie file.....	7
5.8	Installation.....	7
5.9	Supported platforms.....	7
6	What will not be in version 1.0.....	7
6.1	Secondary structure editor	7
6.2	Secondary structure display	7
6.3	Advanced model building.....	7
6.4	Secondary structure figure rendering.....	7
6.5	Molecular dynamics.....	7
7	Documentation.....	7
7.1	Installation guide.....	7
7.2	Tutorial with data examples.....	7
7.3	Users guide describing every button and menu item.	7
8	Quality Assurance.....	7
8.1	Test cases:	7
8.1.1	Robustness to input.....	7
8.1.2	Clustering of rigid bodies.....	8
8.1.3	Unit tests for all computational components, e.g.....	8

8.1.4	Non-inverting transformation	8
8.2	Test output	8
9	References.....	8
10	Acknowledgments.....	8

1 Purpose

This document describes a proposed computer program that will permit the user to generate and view a series of intermediate molecular structures that form a pathway from one conformation of a molecular structure to another conformation of the same molecule. This will allow scientists to investigate the kinetic barriers in macromolecular folding pathways. This tool, Tornado, will also demonstrate the construction of a specialized user interface from a collection of reusable open-source components.

2 Introduction

The long term vision or goal of this work is to identify the highest energy intermediate structure(s) on the lowest energy pathway from one molecular structure to a different conformation of the same molecule. If this lofty goal could be achieved, we would be able to reconcile experimental kinetic folding data with our computational simulations.

The short term achievable goal is to generate movies of the transition from one structure to another.

One implementation of this concept can be found at Yale University:
<http://www.molmovdb.org/morph/>

An intermediate goal is to be able, with manual intervention, to model larger structural changes by manually specifying intermediate structures along the pathway.

3 Issues to be resolved

The following issues must be resolved before SimTK version 1.0:

3.1 *How much does this project overlap or extend the morph server project by Gerstein's group at Yale?*

The novelty of the proposed tool must be evaluated because of the existence of the morph server at Yale.

The morph server uses the not-so-open program CNS to perform its adiabatic interpolation. It is unlikely that we will use CNS as part of the SimTK toolkit. Thus it is likely that we will need to use other molecular mechanics functionality to achieve the same result. Molecular mechanics tools are definitely a specific goal of SimTK, but it is unclear if the SimTK core molecular mechanics tools will be mature enough for this purpose in SimTK 1.0.

3.2 Can we collaborate with Mark Gerstein, Leslie Kuhn, Michael Thorpe, and/or Woolf to accumulate these tools?

Before launching into a significant development effort, we need to evaluate two outcomes: 1) How much novel utility is there for us to create such a tool at SimTK.org, and 2) Can we bring together existing tools to make this happen more easily?

There is some utility in creating a free-er version of tools that already exists, but this does almost nothing to address a research professor's bottom line: novel publications. There needs to be new science in addition to creating tool availability. That is one topic that needs to be addressed before finalizing plans.

4 Users

Target users include scientists interested in transitions from one structure to another. Initial functionality will specifically address the needs of RNA and myosin researchers, because those areas are the initial focus of the Simbios project.

5 Functionality

5.1 User Interface:

The primary interface will be a single window with a motif-style menu bar under the menu bar. On the Macintosh, the motif-style bar will instead be a menu bar on the top of the screen. The menu bar will include the following top level menus: File, Animate, Help. Below the menu bar

5.2 Input:

Start and end structures to be interpolated. The program must work with RNA structures, as well as with protein structures.

5.3 Alignment

In order to interpolate between structures, sets of corresponding atoms must be identified. In the simplest case, there is a one to one correspondence between atoms in the start structure and atoms in the end structure. This case will be handled first. In the most general case, the alignment problem could become quite difficult. Tackling of this problem will be postponed until other elements of this program are in working order. Steps toward tackling the alignment problem might include:

1. elimination of atoms that cannot be aligned
2. accommodation of unalignable atoms by restraining them to maintain similar spatial relationships to their alignable neighbors

5.4 Interpolation

The interpolation between two structures can be accomplished by measuring structure differences at least four levels of granularity. Tools for interpolation at each of these four scales will be developed simultaneously. The four scales, in order of coarsest to finest are:

5.4.1 Global superposition

Global superposition refers to a single rigid body transformation (single global rotation plus translation of all atoms) that relates the atoms of one structure onto another in a manner that minimizes the (weighted) root-mean-squared distances between corresponding atoms. This problem is well solved by Kabsch(1976, 1978), with a useful multiple superposition method provided by Diamond a few years later. The animation interpolation can be computed by interpolating quaternion representations of the rotational component of the transformation, and by direct Cartesian interpolation of the translational component.

5.4.2 Multiple rigid body transformations

Automatically or manually determine large sets of atoms that represent approximately rigid domains that move with respect to one another between the start and end structures, such as that implemented at Gerstein's morph server. This might be accomplished by calculating the transformation for each residue, then clustering those using a measure based on rotation distance (angle between transformations), translation distance (between transformations), sequence distance (between residues), and Cartesian distance (between residues). The goal of this clustering is to reproduce the established number of rigid bodies described by structural biologists.

5.4.3 Torsion angle interpolation

Generate interpolated structures using the torsion angle internal coordinate representation of the structure. Each interpolated structure will be computed by interpolating the torsion angle differences between the start and end structure.

5.4.4 Cartesian interpolation

Cartesian interpolation refers to interpolation between the two structures by moving each atom independently from its start position to its end position.

The first transformation, global superposition, might be applied before the trajectory begins, in order to place the begin- and end-structures in the same reference frame. Next, a combination of domain-scale rigid body transformations with torsion angle rotations could transform the begin structure to closely resemble the end structure. Finally, atom-scale Cartesian interpolation can be used to complete the transformation to the exact end structure. These different scales of transformation can be applied sequentially or simultaneously. The coarser scales, having fewer parameters, simplify the explanation of the transformation, by accomplishing most of the required motions using a relatively small number of parameters.

The four scales of interpolation can be combined by implementing the finer scale interpolations within the implicit reference frame of the coarser interpretations. For example, in a torsion angle interpolation, we might also adjust the relative Cartesian positions of atoms within the rigid groups, such as the six-member side chain ring of phenylalanine.

5.5 Measure energy

At first, just measure the energy of the intermediate structures by any of a variety of means. For example, compute the energy using a force field from Gromacs.

Optionally view the atoms that are involved in collisions by color coding them according to the severity of the collision.

Eventually, the goal is to generate low energy pathways between the start and end structures. Steps on this pathway include:

1. Generate intermediate “key” frames that help avoid collisions and other high energy problems.
2. Include dynamics or kinematics in the sequence of interpolations to explore low energy pathways between the start and end.

5.6 Relax

At each step in the transition, the total energy of the molecule can be relaxed by energy minimization. It may be necessary to use restraints to bias the result toward the final structure and away from the initial structure.

5.7 Output

The following outputs will be available:

5.7.1 Coordinate files

A sequence of atomic structure files corresponding to frames along the pathway from the starting structure to the end.

5.7.2 Parameter trajectories

A sequence of parameters that were interpolated, possibly including:

1. atomic coordinates, in both local and global coordinate systems
2. transformations, including translations, rotations in both matrix and quaternion form
3. Computed energy at each time point. This should be reported in a tab-delimited format, so that graphs can be easily made.

5.7.3 Movie files

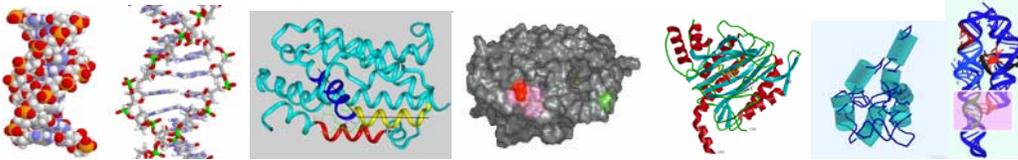
Movie files will be created using an interactive trajectory editor. The editor will have playback features, analogous to a VCR.

1. buttons:



- a. forward/play >
- b. pause ||

- c. reverse <
 - d. step forward |>
 - e. step back <|
 - f. go to beginning |<<
 - g. go to end >>|
 - h. Mouse over (hover) on button reveals tool tip button description, e.g. “Play in reverse”, “Step forward one frame”, etc.
 - i. each button will have a short text name on the icon, e.g. “forward”, “reverse”, “beginning”, “end”, “step”, “step back”, “pause” (not shown in mock up above)
 - j. speed control (appearance TBD)
2. near button panel is a slider the full width of the window
 - a. Dragging and clicking the slider sets the view to a particular frame of the animation, using as much of the interface of a standard scroll bar as possible.
 3. near the buttons and slider is a text presentation of the frame number and total frame count, for example “Frame 64/128”
 4. user adjustable orientation, zoom, translation
 - a. mouse drag in graphics window, modified by alt, ctrl, shift; and other mouse buttons
 5. optional automatic view down hinge axis or particular torsion angle
 6. Ability to display different representations for different parts of the molecule. These features will be derived from reusable open-source tools, which will be reusable in other specialized user interfaces.
 - a. selection syntax at
 - i. atom
 - ii. residue
 - iii. molecule
 - iv. secondary structure level
 - b. representations:



- i. Space filling – One sphere centered on each atom; sphere radius is atom’s van der Waals radius
 - ii. bond sticks
 - iii. backbone trace
 - iv. molecular surface
 - v. Richardson ribbons
 - vi. duplex cylinders
 - vii. nucleotide sticks
7. Allow the user to specify the camera parameters (rotation, zoom, translation) at points in time, and automatically interpolate these.

5.7.4 Movie file

Save As->some reasonably widespread movie format (to be identified)

5.8 Installation

- a. Java web start – click on URL

5.9 Supported platforms

- b. Windows XP 32 bit
- c. Linux x86 32 bit
- d. Linux x86 64 bit
- e. Mac OS 10.4

6 What will not be in version 1.0

6.1 Secondary structure editor

6.2 Secondary structure display

6.3 Advanced model building

6.4 Secondary structure figure rendering

6.5 Molecular dynamics

7 Documentation

The following documentation will be needed:

7.1 Installation guide

7.2 Tutorial with data examples

7.3 Users guide describing every button and menu item.

- a. HTML version accessible from “Help” menu (Help->User Guide)

8 Quality Assurance

8.1 Test cases:

8.1.1 Robustness to input

The program must be able to load, without failure, each of the structures in the protein data bank. Further, the program must respond gracefully to attempts to load non-structures, including binary files and empty files.

8.1.2 Clustering of rigid bodies

For structures with an established hinge motion of one or more components, the multiple rigid body superpositions must be able to match the number of rigid bodies, with 90% accuracy on the set of residues involved in each body.

8.1.3 Unit tests for all computational components, e.g.

8.1.4 Non-inverting transformation

Single rigid-body transformation must not invert coordinates, even when the target positions are a mirror image of the starting position. In other words, the determinant of the rotation matrix must be positive.

8.2 Test output

Each test must be automated with a “pass” or “fail” result reported.

9 References

Diamond, R. (1988). A note on the rotational superposition problem. *Acta Cryst.*, A44, 211-216.

Diamond, R. D. (1992). On the Multiple Simultaneous Superposition of Coordinate Sets by Rigid-Body Transformations. *Protein Science* 1(10): 1279-1287

Horn, B.K.P., “Closed-Form Solution of Absolute Orientation using Unit Quaternions,” *Journal of the Optical Society A*, Vol. 4, No. 4, pp. 629–642, April 1987.

Kabsch, W. (1976) A solution for the best rotation to relate two sets of vectors. *Acta Crystallogr.*, A32, 922–923.

Kabsch, W. (1978) A discussion of the solution for the best rotation to relate two sets of vectors. *Acta Crystallogr.*, A34, 827–828.

WG Krebs, M Gerstein (2000) The morph server: a standardized system for analyzing and visualizing macromolecular motions in a database framework. *Nucleic Acids Res* 28: 1665-75.

10 Acknowledgments

This work was funded by the National Institutes of Health through the NIH Roadmap for Medical Research Grant U54 GM072970.

Information on the National Centers for Biomedical Computing can be obtained from <http://nihroadmap.nih.gov/bioinformatics>.