Blender for biology: Game Engine for molecular animation and special effects for chemical and physical behavior

Zini MF[1], Andrei R[1,2], Loni T[3], Cianchetta S[1], Callieri M[4], Zoppè M[1]

Scientific Visualization Unit

[1]Institute of Clinical Physiology, CNR; [2]Scuola Normale Superiore; [3]Big Bang Solutions (Navacchio); [4]ISTI – CNR. Pisa, Italy.

www.scivis.ifc.cnr.it

INTRODUCTION

The fact that proteins move continuously in the cellular environment has finally overcome the concept of structure-function, which tacitly implied a single conformation for any given molecule. The features of the movements, however, are still difficult to calculate and represent, in particular for practicing laboratory (wet) biologists, who are the people that would most benefit form an easy visualization of such motions.

Beside the change in conformation, resulting in different shapes of the molecules, relative motion of the aa composing the surface of the proteins confer dynamic properties to the surface, such as the exposition of hydrophobic patches, or availability of charged spots.

In an effort to address both these aspects (motion and visualization) of moving proteins, we have implemented a series of python scripts that work in Blender, an open source program of 3D animation, visual effects and video games.
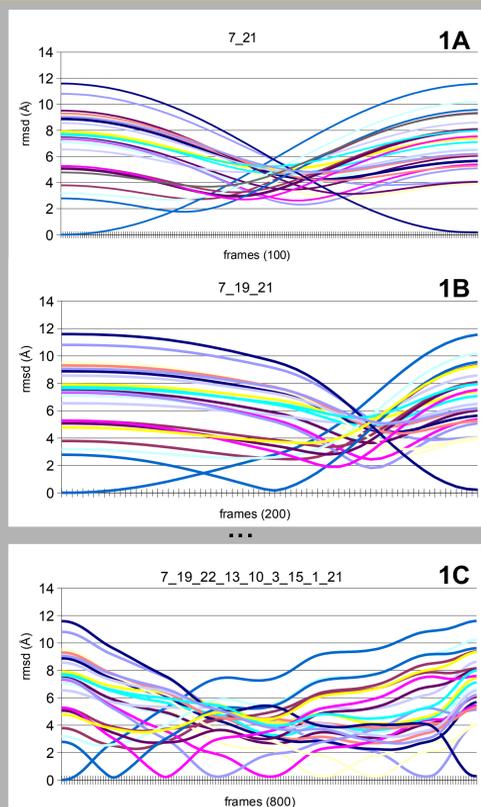

Fig. 1.Organizing NMR models to obtain a smooth transition that visits several conformations. The first two steps and the last are shown.
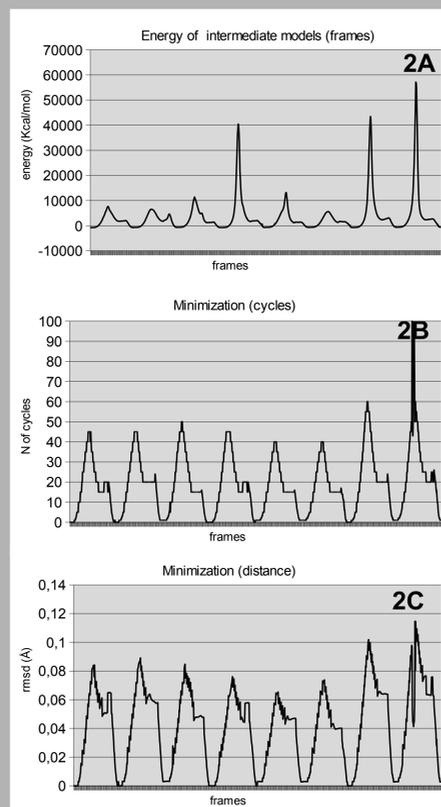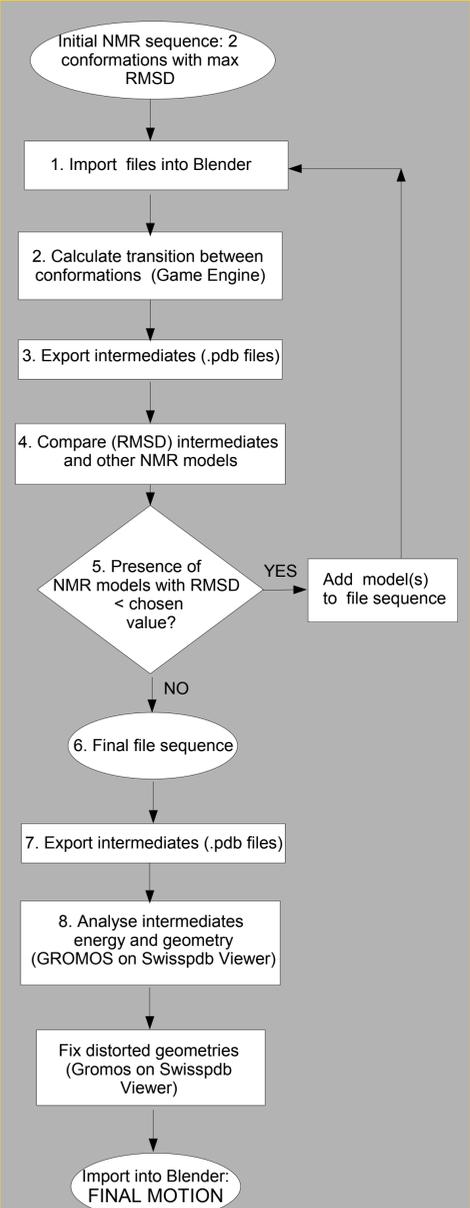
PROCEDURE

Two sequential automated procedures have been followed to navigate an NMR collection (exemplified with PDB file 1cfc, apoCalmodulin[1]) in a fast and efficient way, and produce a continuous motion of the protein.

As illustrated in the diagram the file is first analysed to identify the two conformations with the highest RMSD. These are imported as the starting point in Blender[2], and its Game Engine is allowed to calculate intermediate conformations considering both the links between atoms (rigid body joints allowed only rotation along their own axis) and collisions between all atoms (step 2).

Interpolated conformations are exported as pseudo pdb files and compared with all other models in the original NMR collection (steps 3 and 4, fig. 1A). NMR models with RMSD lower than 2 Å relative to any of the interpolated conformations are inserted (5, fig. 1B and 1C), and the procedure is reiterated. In this way 9 models of the 1cfc NMR of Calmodulin were ordered for a smooth motion (result 6, Final file sequence).

The following steps are automated elaboration of each single frame obtained in the previous part: the interpolated positions of the protein are analysed using GROMOS[3] force field in SwissPDB-Viewer[4], and adjusted if necessary. Fig.2A shows the energy content of all models calculated by Blender GE. The major component that contributes to the peaks are confined to only few intermediates and are due to non bonded clashes (most often artefacts related to rotameric conversions). To adjust every intermediate we set as threshold a level of energy comparable with the energy of experimental NMR conformations, and allowed GROMOS to run the necessary number of minimization cycles to reach it. Most conformations reached the requested energy with a very low N of cycles (<50, Fig. 2B), and with very minor displacement of atoms (max RMSD between before and after minimization <0.12 Å, as shown in panel C).

Fig. 2. Geometrical adjustment and energy minimization of each interpolated conformation elaborated by Blender GE

From these studies we conclude that properly equipped Blender Game Engine can find new conformations of proteins as transition forms between a set of given experimental models. These are easily adjusted to models compatible with physical and chemical constraints. The reiteration of the procedure proposed can be repeated to include conformations not originally identified as continuum of dynamical sequence. The system is fast, easy and available for free.

PROCEDURE FOR RENDERING

Once obtained a sequence of smoothly changing conformations, it should be possible for scientists to visualize the molecules in a way that directly conveys also information about their surface properties. To this aim, we have set up a rendering procedure that shows simultaneously the electric and lipophilic potentials, calculated frame by frame.

Each file (frame) from the procedure outlined above is exported as .pdb and elaborated according to the scheme shown in the diagram.
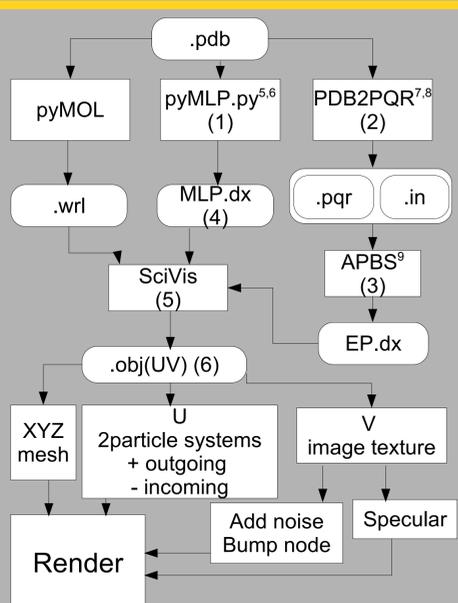
(1) Python script that using atomic lipophilicity values calculates lipophilic potential in every point of the grid surrounding the protein.

(2) Software package that attributes atomic lipophilicity values and prepares structures for continuum electrostatics calculation. http://pdb2pqr.sourceforge.net/

(3) APBS calculates Electrostatic Potential by solving the Poisson-Boltzmann equation. http://apbs.sourceforge.net/

(4) Molecular Lipophilic Potential is calculated according to Testa's formula[10].

(5) With a home made program the values of EP and MLP are mapped on the surface of the protein, so that each vertex of the mesh is associated a value of each potential.

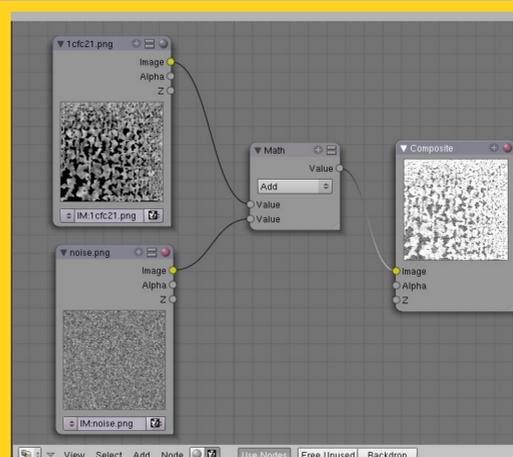(6) The file is now imported again in Blender where all subsequent elaborations are performed.

Fig.4 Example of rendering node in Blender. On the left the image obtained from the V surface mapping of lipohilic values (top) and the bump map (bottom); the mixing node and the final texture to be mapped on the protein surface

The visual properties that we propose as a code for physical and chemical surface features are immediately decoded by humans, because they correspond to textures of real material (MLP) and to familiar phenomena as smoke (EP) allowing a better perception of molecule's dynamic behavior.
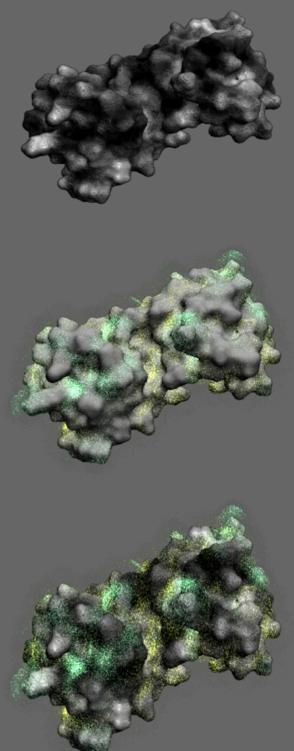

Fig. 3. Surface features of Calmodulin. Top, MLP: bump and dark texture for more hydrophilic, more light and smooth for lipophilic areas. Middle, EP: Positive values (rendered as dynamic incoming particles in movie) are colored green, Negative values (ougoing) are yellow. Bottom, combination of the two features

REFERENCES

1. Kuboniwa, H. et al. Solution structure of calcium-free calmodulin. Nature Struct. Biol. 1995, 2, 768–776
2. www.blender.org
3. van Gunsteren, WF. et al. in Biomolecular simulation: the GROMOS96 manual and user guide. Vdf Hochschulverlag ETHZ,1996.
4. Guex, N. et al. SWISS-MODEL and the Swiss-PdbViewer: An environment for comparative protein modeling. Electrophoresis 1997, 18, 2714-2723.
5. Broto P. et al. Molecular structures: Perception, autocorrelation descriptor and sar studies. System of atomic contributions for the calculation of the n-octanol/water partition coefficients, Eu. J. Med. Chem. 1984, 19.1, 71-78

6. Laguerre M. et al. MLPP: A program for the calculation of molecular lipophilicity potential in proteins, Pharm. Sci. 1997, 3.5-6, 217-222
7. Dolinsky T.J. et al. PDB2PQR: Expanding and upgrading automated preparation of biomolecular structures for molecular simulations. Nucleic Acids Res 2007, 35, 522-525.
8. Dolinsky T.J. et al. PDB2PQR: an automated pipeline for the setup, execution, and analysis of Poisson-Boltzmann electrostatics calculations. Nucleic Acids Res 2004, 32, 665-667.
9. Baker NA. et al. Electrostatics of nanosystems: application to microtubules and the ribosome. Proc. Natl. Acad. Sci. USA 2001, 98, 10037-10041.
10. Gaillard P. et al. Molecular lipophilicity potential, a tool in 3D QSAR: Method and applications. J Comput-Aided Mol Des 1994, 8, 83–96.