11-2014

Automatic Animation of Molecular Motion using Python and Cinema 4D

Diana Zajac
DePauw University

Nathaniel Smith
DePauw University

Dan Gurnon
DePauw University, danielgurnon@depauw.edu

Automatic Animation of Molecular Motion using Python and Cinema 4D

Diana Zajac, Nathaniel Smith, Professor Dan Gurnon

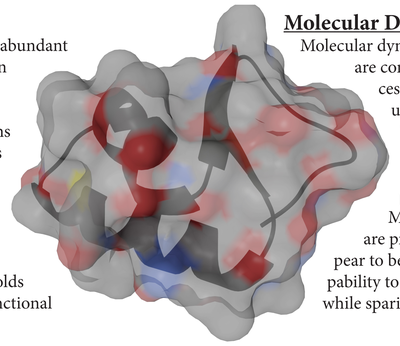
Department of Chemistry and Biochemistry, DePauw University

Summer, 2014

Background Information:

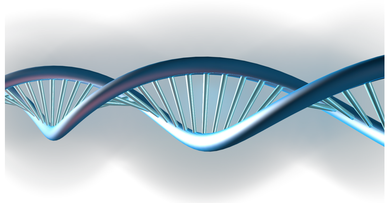
Proteins

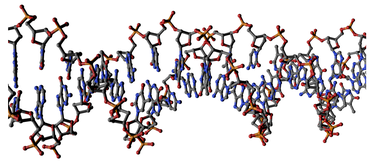
Proteins are incredibly abundant molecules in the human body that act as machines to carry out cellular processes. Proteins are composed of chains of 20 different amino acid building blocks strung together like beads on a necklace. Once synthesized, this "molecular necklace" folds to make a dynamic, functional structure.



Molecular Dynamics

Molecular dynamics (MD) simulations are computationally-intensive processes that can predict molecular movement and behavior. These processes are very complicated and require days of calculation on a supercomputer. However, "fake Molecular Dynamics", which are prescripted dynamics that appear to behave randomly, have the capability to achieve similar visual effects while sparing the resources for real MD.

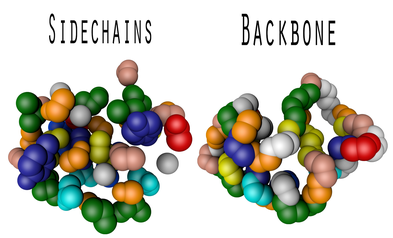




Inspiration:

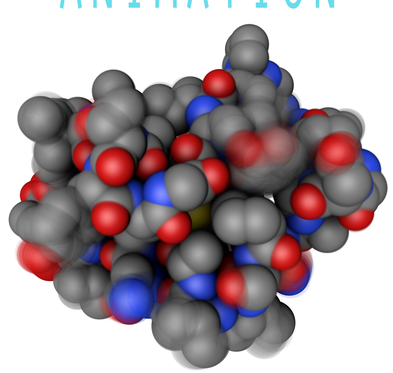
Science and art have been separate disciplines for centuries, but there is a mutualistic connection between creativity and scientific knowledge. Portraying science through art not only inspires aspiring scientists, but helps everyone visualize the complicated molecular world.

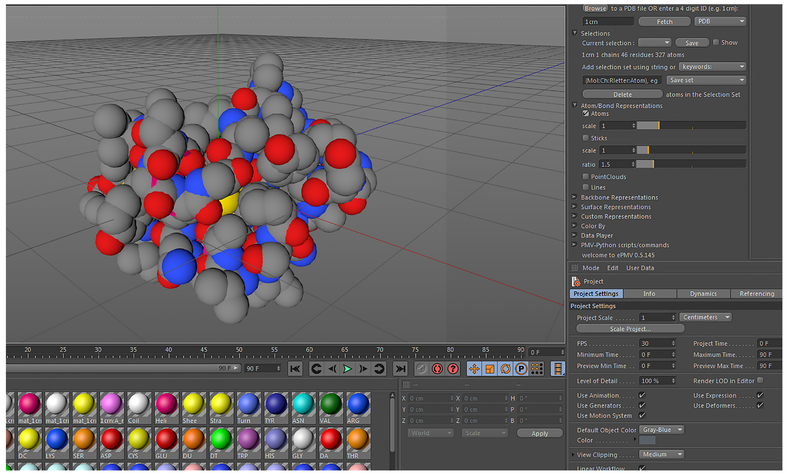
Molecular animation poses a challenge - there is great potential for molecular animation as an educational tool, yet most animators are not trained in the science of molecular motion. Although animators must simplify the science to communicate effectively, oversimplification can be misleading.



Goal:

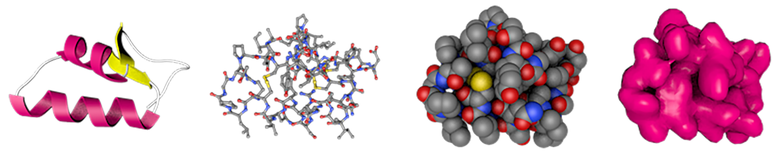
The goal of our project was to give animators access to resources that would allow them to bridge the gap between the accuracy of science and the creativity of digital art. Specifically, we wanted to script the science so that molecular animators could automatically rig a protein to look and move like a molecular dynamics simulation. We wanted to design a free and open-source script that would be easily accessible to anyone. Then, with the click of a single button, animators would be able to transform their rigid molecules into rigged motive molecules. The transformed molecules would animate based on the bonding between atoms.





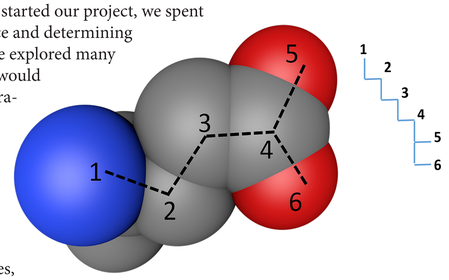
Resources:

Our project would not have been possible without a multitude of programs and tutorials to assist our animations and script-writing. Maxon's Cinema 4D is a 3D animation software that was the platform for our molecular animation. To learn how to use Cinema 4D, we utilized GreyScaleGorilla's Cinema 4D beginner tutorials (greyscalegorilla.com). Within Cinema 4D, we also used a plugin called the embedded Python Molecular Viewer (ePMV) to import and display protein data bank files (http://epmv.scripps.edu/). When scripting, we learned the basics of Python from Codecademy (codecademy.com), and we directed specific coding questions to Cineversity's open forum (cineversity.com). We also used Adobe After Effects CC to render our videos and Jing screencapture software to create our tutorials.



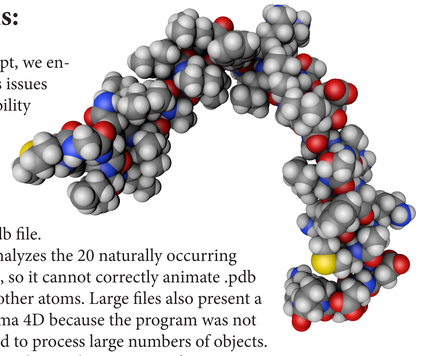
Method:

We spent a significant amount of time learning how to use Cinema 4D and write in the Python scripting language. When we started our project, we spent four weeks learning the Cinema 4D interface and determining how to manually rig a protein molecule. We explored many methods of achieving random motion that would not be specific to Cinema 4D, including vibration, turbulence, random forces, and effectors. Our method was intended to be easily accessible in other programs, despite differences in specific features. The major parts of our script involved: isolating atoms from the .pdb file, categorizing and grouping the backbone and sidechain atoms, creating amino acid-specific hierarchies, making fracture objects to house the atom groups, and assigning effectors to the appropriate atom groups.

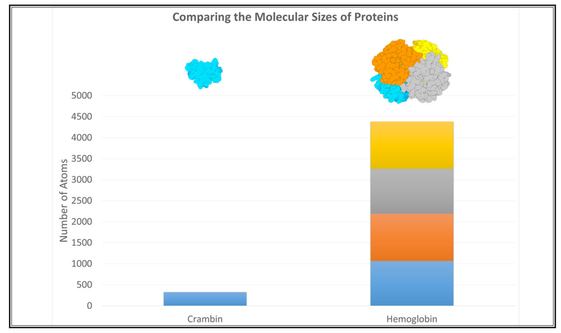



```
for cb_op in cb_list:                    #Makes nulls for every amino acid sidechain
    cb_null = c4d.BaseObject(c4d.Onull)
    cb_name = cb_op.GetName()
    cb_null.SetName(cb_name.split(".")[3])
    cb_null.InsertUnder(SCfrac)
    cb_null.SetMg(cb_op.GetMg())
    GroupAtoms(doc.GetActiveObject(),cb_name.split(".")[3],cb_null)

for bb_op in bb_list:                    #Inserts backbone atoms under the backbone fracture
    mg = bb_op.GetMg()
    bb_op.InsertUnder(BBfrac)
    bb_op.SetMg(mg)

pro_list = FindP(op)                     #Adds proline atoms back into the sidechain fracture
for pro in pro_list:
    pro.InsertUnder(SCfrac)


fracList = AA_hierarchy(op)              #Runs the hierarchy script
```

Limitations:



In testing our script, we encountered various issues that limit the usability of our code. One limitation is our script's inability to process hydrogen atoms if they are in the .pdb file. Our script only analyzes the 20 naturally occurring amino acid atoms, so it cannot correctly animate .pdb files that contain other atoms. Large files also present a problem for Cinema 4D because the program was not originally intended to process large numbers of objects. In order to avoid crashes in the viewport of Cinema 4D, we recommend separately rigging each chain of a large multimer.



Fall Research and Future Research:

Beyond vibration and rotation, there is a third tier of molecular motion - flexibility in a molecule's backbone structure. Proteins have flexibility in their secondary structures, so a complete representation of protein dynamics should include all three tiers of motion. This fall, we focused on simulating backbone motion and unfolding of alpha helices. We successfully identified a method of creating this motion, but we have not yet automated this process.

To further this research, we will continue to make new tools to automate the process of protein animation, with the overall aim of making it easy to be accurate. All of our scripts will be free and open-source for other animators to use and improve upon.

References:

Johnson, GT, L. Autin, D. S. Goodsell, M. F. Sanner, and A. J. Olson. "EPMV Embeds Molecular Modeling into Professional Animation Software Environments." Structure 19.3 (2011): 293-303.

McGill, G. "Molecular Movies... Coming to a Lecture near You." Cell 133.7 (2008): 1127-32.

PDB ID's used to generate images on this poster: 1CRN (Crambin) and 2LZR (Twin Arginine Translocase)

DNA PDB file was generated using the w3DNA interface at http://w3dna.rutgers.edu/

Acknowledgments:

Funding provided by the Science Research Fellows Program and DePauw University