User documentation for DROIDS 1.20 – a GUI-based pipeline for comparative protein dynamics

Gregory A. Babbitt[1*], Jamie S. Mortensen[2], Erin E. Coppola[2], Lily E. Adams[1], Justin K. Liao[2]

1. T.H. Gosnell School of Life Sciences, Rochester Institute of Technology, Rochester NY
2. Biomedical Engineering, Rochester Institute of Technology, Rochester NY

*corresponding author email address: gabsbi@rit.edu

Overview

DROIDS 1.20 is an open source software project aiming to visualize and quantify the impact of one of the longest time scale processes in the universe (i.e. molecular evolution) on one of the shortest time scale processes in the universe (i.e. molecular motion). Specifically, we want to know how molecular evolution over 100s of millions of years impacts the functional molecular motions that play out over a few femtoseconds in real time. A primary motivation of this project is to combine GPU accelerated biophysical simulations and GPU graphics to design a gaming PC into a 'computational microscope' that is capable seeing how mutations and other molecular events like binding, bending and bonding affect the functioning of proteins and nucleic acids. DROIDS-1.20 (**D**etecting **R**elative **O**utlier **I**mpacts in molecular **D**ynamic **S**imulation) is a GUI-based pipeline that works with AMBER16, Chimera 1.11 and CPPTRAJ to analyze and visualize comparative protein dynamics on GPU accelerated Linux graphics workstations. DROIDS employs a robust and nonparametric statistical method (multiple test corrected KS tests on all backbone atoms of each amino acid) to detect significant changes in molecular dynamics simulated on two homologous PDB structures. Quantitative differences in atom fluctuation (i.e. calculated from vector trajectories) are displayed graphically and mapped onto movie images of the protein dynamics at the level of individual residues. P values indicating significant changes are also able to be similarly mapped. DROIDS is useful for examining how mutations, epigenetic changes, or binding interactions affect protein dynamics. DROIDS was produced by student effort at the Rochester Institute of Technology under the direction of Dr. Gregory A. Babbitt as a collaborative project between the Gosnell School of Life Sciences and the Biomedical Engineering Dept. Visit our lab website (https://people.rit.edu/gabsbi/) and download DROIDS from Github at https://github.com/gbabbitt/DROIDS-1.0. We will be posting video results periodically on our youtube channel https://www.youtube.com/channel/UCJTBqGq01pBCMDQikn566Kw

A single page Quick Start Guide (pdf) and full Installation Guide and User Manual are available with the download. They outlines the processes encountered in each of the three GUI interfaces. It is strongly advised that users be comfortable with how to prepare PDB files for molecular dynamic (MD) simulation using GPU accelerated AMBER 16 (pmemd.cuda). DROIDS assumes that .pdb files named in the GUI have been properly modified for AMBER simulation. You must consult the AMBER documentation for this step. The DROIDS GUI provides automation of teLeap, a program for pdb file setup, but care must be taken to read output on the Linux terminal for any errors. The programs 'antechamber' and 'pbd4amber'

are also helpful in modifying files for MD and may be necessary to run manually prior to starting teLeap in DROIDS. Please consult the Amber16 user manual for more details. Typically preparation includes adding hydrogens, removing mirrored images, removing waters and other chemical artifacts, loading information regarding important ligands. teLeap does much of this automatically, however the user is responsible for carefully reading the teLeap output at the terminal for any indications of problems. ALSO NOTE: AMBER 16 software must be licensed from the University of California. More details about purchasing and installation can be found at http://ambermd.org/. DROIDS is tested on Linux Mint 18.1 and Ubuntu 16.04 and is offered freely under the GPL 3.0 license and is available on GitHub https://github.com/gbabbitt/DROIDS-1.0

Primary software dependencies are Amber16, Ambertools16 or 17, CUDA 8.0, UCSF Chimera 1.11 (or higher), perl-tk (perl 5.10), python-tk, python-gi, R-base, R-dev, ggplot2 and gridExtra (R packages), evince (Linux pdf viewer), GStreamer (Linux movie viewer) and descriptive.pm (a perl statistical module provided with our download). Hardware dependencies are only to have a high end Nvidia graphics card with proper drivers. All testing of DROIDS 1.0 was done on the GeForce GTX Titan X and GTX 1080 GPUs. Linux versions of Chimera are available at (https://www.cgl.ucsf.edu/chimera/). All other dependencies are able to be addressed via the usual Debian package downloads. CUDA drivers for Linux are available from Nvidia. NOTE: do not use CUDA 7.5 with Amber16. CUDA 8.0 is currently supported.

The supporting scripts 'teLeap_proteinReference.pl' and 'teLeap_proteinQuery.pl' have hard coded paths to the teLeap program binaries and the force field library. As Amber16 is typically installed to the Desktop, this path will be different on different machines. Make sure you edit the path appropriately before attempting to run DROIDS. Also, DROIDS has intended that Chimera 1.11 has been installed using the default path. All of the scripts that utilize Chimera will have the path hard coded at the top of the script. Alter appropriately if your version of Chimera uses an alternate path.

The DROIDS pipeline

The DROIDS pipeline is run as a series of 3 linked Perl scripts that are controlled at the command line. The Quick Start Guide lists the steps shown schematically in Figure 1. The user starts the pipeline by placing the two PDB files to be compared in the DROIDS main folder, opening a terminal, and typing 'perl GUI_START_DROIDS.pl. We also offer an alternative GUI for computer builds with dual GPU cards (GUI_START_DROIDS_dualGPU.pl). This will run MD on both homologous protein structures at the same time. This GUI interface is designed to control and run all stages of the MD simulations of both the query and reference PDB structures that will be needed for later DROIDS analysis. This includes typical teLeap setup of the PDB file, structural alignment of the query and reference proteins, and an energy minimization, heating and equilibration run on each PDB. These runs are followed by N number of sampling runs with N specified by the user. Random spacer runs precede each sampling run so as to

minimize the impact of initial conditions on the MD sampling (i.e. minimize differences merely due to chaos in the MD runs). Afterwards, a GUI for vector trajectory analysis will popup. This leads the user through typical cpptraj commands to collect atom fluctuation and correlation data for each sampling run. Lastly, this GUI leads the user through data preparation for later DROIDS statistical analysis. This step includes the parsing of the vector trajectory output to the structurally-based sequence alignment in performed earlier in Chimera. This is followed by choice of 'strict' vs 'loose' homology (which determines upon which amino acids the DROIDS statistics will be applied). Loose homology should be chosen when evolutionary distances are large. Strict homology should be chosen when sequences are nearly identical (e.g. examination of one or several mutations). The third GUI allows users to run the statistical comparisons and choose method of multiple test correction, followed by color and graphics options to be applied to the static and moving images of the reference PDB.

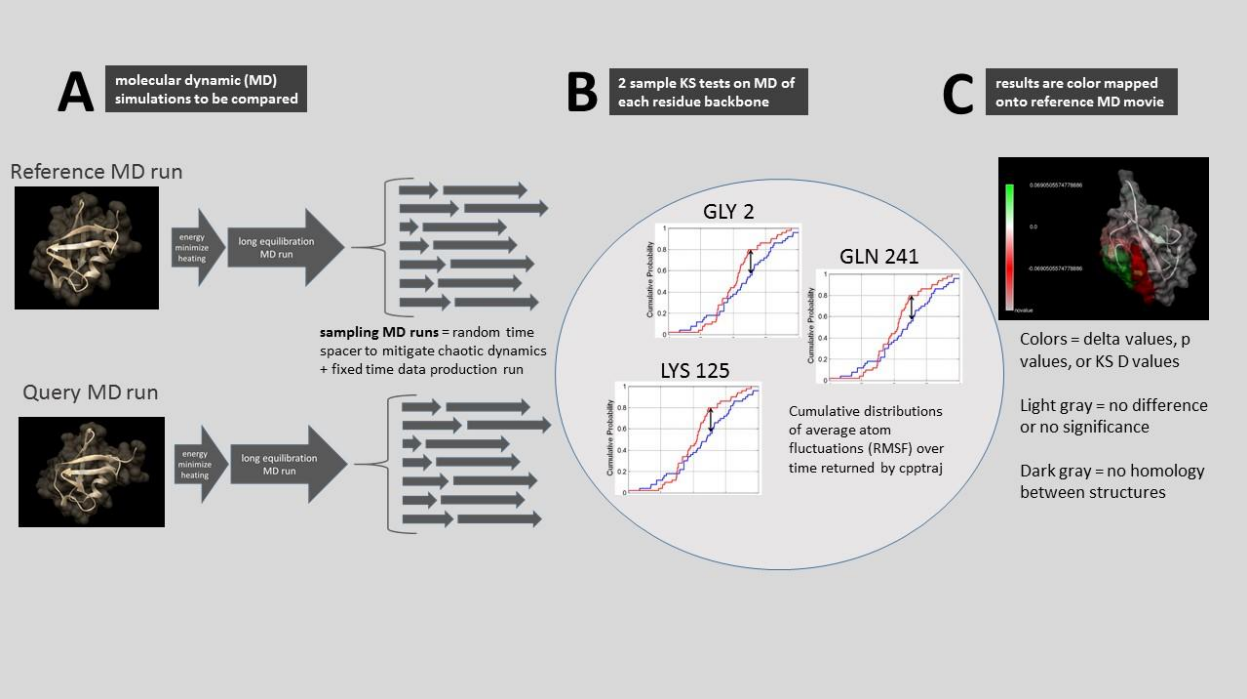

Figure 1. A schematic representation of DROIDS comparative molecular dynamic analysis software. DROIDS 1.20 is a software tool for multiple test corrected amino acid-level pairwise comparison of molecular dynamics of two comparable PDB structures. The three main phases of analysis include (A) MD sampling runs and vector trajectory analysis, (B) statistical comparison via multiple test corrected KS tests, and (C) visualization results on static and moving images.

The statistical test is a KS test applied specifically to the collective backbone MD of each amino acid residue (i.e. atoms N, CA, C and O masked during cpptraj).

GUI_START_DROIDS.pl.

This first GUI interface allows the user to set the most important parameters for the MD (e.g. name the force field, set run times of each phase, choose a solvation method, add salt conc) as well as determine how many sampling MD runs on each protein will be analyzed in later analysis. For most proteins, I often take 50-100 sampling runs at 0.5ns each, after a single equilibration phase of 10-50ns…depending upon how stable the structure behaves. Users are guided through creation of a structurally-based sequence alignment using Chimera MatchMaker and Match->Align, followed by setup of topology and coordinate files using teLeap. Then the script automates the energy minimization, heating, equilibration and MD production sampling runs on the two homologous structures and reports the progress to the Linux terminal. This part of the analysis takes the longest (e.g. the two comparative runs on two typical implicitly solvated systems may take 24-48 hours to run on the GTX 1080 card). Explicit solvated systems may run 2-3X longer. Details about the MD are hard coded into the portion of the script that writes the control file (i.e. the control subroutine). These settings can be easily changed by users with some experience with Amber commands and perl scripting. The default assumes constant temperature (300K) and pressure during production. Note that MD output is produced in the form of binary files (.nc file type extension) rather than text (i.e. .mdcrd file type). This is to allow the saving of hard drive space and proper file type for cpptraj analysis that follows. These files are not 'readable' in any sort of text editor. Jobs are scheduled to the GPU by means of a while loop that periodically pgreps the process ID's produced by pmemd.cuda. The GPU will not automatically control job scheduling the way a CPU will. So we have added a GPU surveillance button that opens terminals that monitor the load on the GPU as well as current running processes. If the user interrupts a script and starts another job, this will not terminate the previous run. If the user sees that two pmemd.cuda processes are running at once, then the data is likely corrupt as the GPU is attempting to run both jobs at the same time. We include a 'kill' button which will pkill all pmemd.cuda jobs. This is handy when restarting DROIDS after previous interruption. It is recommended that user keep surveillance open at all times alongside the main terminal when running then MD wrapping script (GUI_START_DROIDS.pl). See Figure 2 for how this should look on your desktop. Before each sampling MD run, a random time length spacer is generated uniformly distributed between 0 and 0.5 x length of the sampling run. The purpose of this step is to average out the effect of chaotic dynamics that may be observed if the initial starting conditions were always exactly taken after the equilibration step has finished. A typical DROIDS analysis might consist of 0.5ns heating, 10-50ns of equilibration and 50 x 0.5ns of sampling runs on each protein. With this setting, most comparisons of protein dynamics can be achieved in 12-48 hours of run time using a dual GPU machine with GTX 1080.

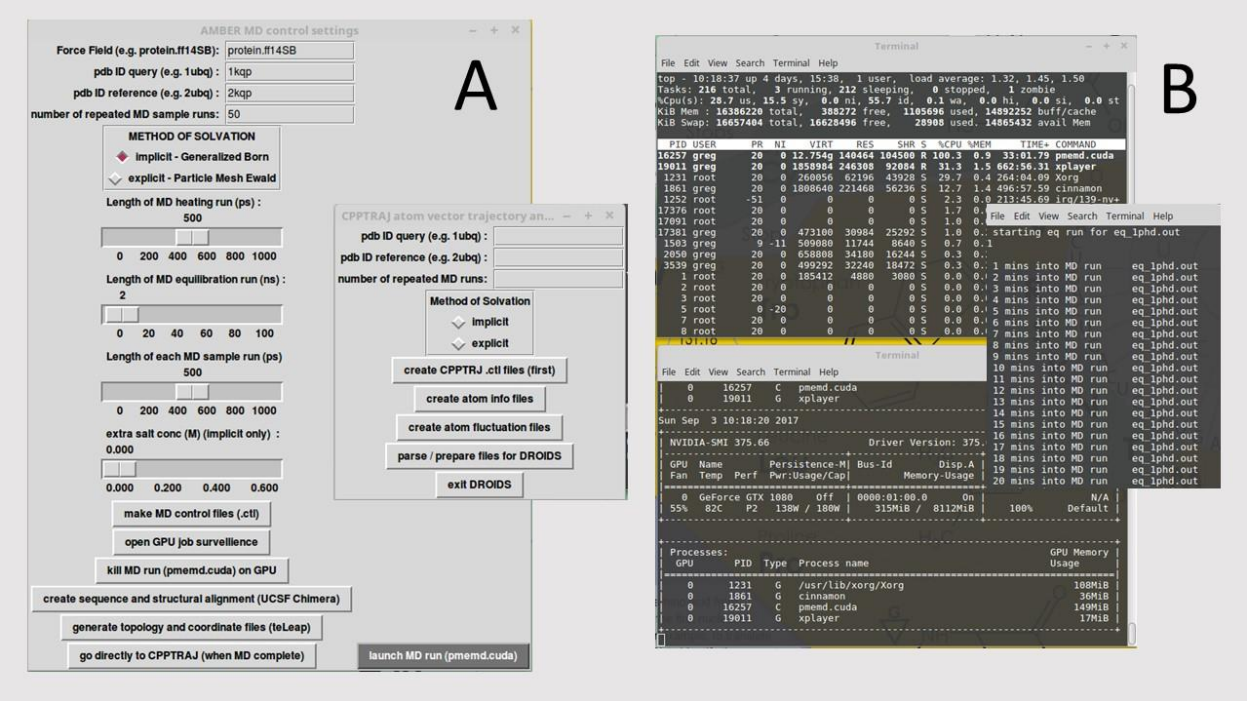

Figure 2. (A) The DROIDS 1.20 GUI interfaces for controlling molecular dynamic simulations and sampling conditions in Amber16 and subsequent cpptraj analysis. (B) Linux terminal windows showing the progression of the MD simulations as well as general surveillance of GPU loads and process IDs.

GUI2_DROIDS.pl

A prompt at the end of the MD simulations will open the next GUI designed to guide the user through vector trajectory analysis using cpptraj (Ambertools16/17). See Figure 2 (inset). The buttons are run from top to bottom and include making control files, collecting atom information, calculating atom fluctuations, and lastly, preparing and parsing the cpptraj output for subsequent DROIDS analysis on GUI 3. The

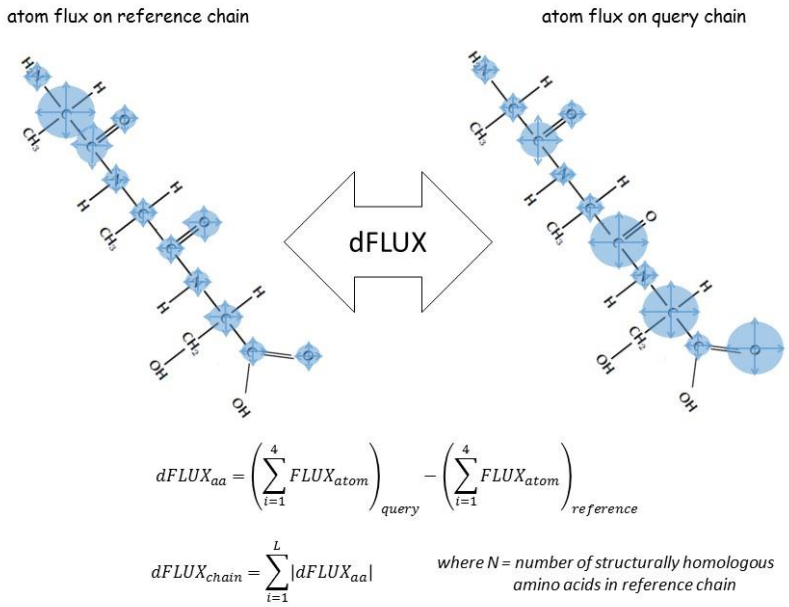

Figure 3. A schematic representation hypothetical differences atom fluctuation (dFLUX). Functional analysis of destabilization due to mutation and or evolution of functional thermostability can be addressed using dFLUX. In the DROIDS color mapping, dFLUX is averaged over the 4 backbone atoms of each amino acid. Global dFLUX for the whole chain is simply the sum of absolute dFLUX over the length of the polypeptide chain.

setup we use under the hood is designed to return amino acid averaged motions collected only over the backbone of the polypeptide chain (i.e. N, CA, C, O). Fluctuation is very rapid (10-20 femtoseconds on most bonds) and largely harmonic and thus is relevant to comparative studies of protein stability (i.e. evolution of thermostability, functional epigenetic modifications, or disease-related genetic mutations that globally destabilize function. During initial setup (start GUI), the user is also guided from the terminal through the creation of a structural alignment of both protein structures using Chimera's MatchMaker and Match -> Align tools. The user is directed to save the resulting sequence alignment as a Clustal format file (.aln) using the name of the reference PDB ID in the title as follows Nxxx_align.aln (e.g. ubiquitin would be 1ubq_align.aln). Not that it is very important that the user trims the chains to the same length after alignment so that data is collected correctly from homologous amino acids. In GUI 2, the user is now also asked to specify whether the DROIDS statistics and mapping are to be conducted using 'loose' or 'strict' homology. Strict homology will only conduct MD comparisons on the backbone atoms of the protein when the aligned amino acid residues are identical. Loose homology will compare backbone MD even when residues are different as long as the structural alignment file identifies then as homologous. Note: atoms in sidechains are always excluded from all analyses via a mask used in cpptraj. Using strict homology on a protein comparison without a large evolutionary distance along with brighter color selections for nonhomologous regions can be a way to label interesting mutations in the resulting images

and movies of the dynamics. Under structural comparisons of greater evolutionary distances, where the underlying protein sequences are likely to be quite different, it is best to select loose homology along with a less conspicuous color (i.e. gray or tan) to mark regions in the protein comparison that lack true homology (i.e. are poorly aligned) MatchMaker provides user ability to choose appropriate substitution matrices and gap penalties to reduce the problem of poor alignment. DROIDS automatically excludes these regions from analysis. NOTE: at the end of parsing, a folder named 'atomflux' should appear with individual files for each comparison per residue. The number of files in this folder should correspond to the number of residues in the reference protein that have homologous residues in the query protein. If there are far fewer files in the atomflux folder than expected, this is most likely due to the fact the sequence at PDB does not exactly match the structure. Occasionally, one will need to trim the alignment file to match the structure, and then rerun the parsing again.

GUI3_DROIDS.pl

This last GUI (Figure 4) is the heart of comparative protein dynamics using DROIDS. The initial steps include making choices about the type of analysis you want, then producing the control files you need. Then you run the KS tests in R on the next button. R graphics will show analyses as a popup in the pdf viewer. After this step the user will generate Chimera 'attribute' files for color mapping. Color mapping generally scales in saturation with the strength of the delta shift in atom motion (fluctuation or correlation) between the two sets of MD runs. Regions lacking homology are darker gray. If you are only changing the mapping options (i.e. data types – delta, p, or D values, color schemes or scaling of plots), you do not need to rerun the statistical tests. If you change statistical test options (i.e. motion type, p value cutoff, or multiple test correction), you will need to rerun the KS tests again. As the number of KS tests equals the number of amino acids on the chain, correction for multiple testing is highly recommended. Multiple test correction methods included as options in DROIDS are the Bonferroni correction or Benjamini-Hochberg estimation of false discovery rate. The dFLUX values of the query runs can be scaled to the absolute dFLUX values of the reference runs if the user is more interested in relative difference rather than absolute difference. Be sure to choose color schemes that correspond to the data type as indicated on the screen. Color gradients can be auto-scaled (highest to lowest value) or fixed at one of several options. When statistical options are changed (excepting p value corrections) a new DROIDS results folder is generated for each set of tests. After the Chimera attributes are stored for mapping, the user can generate color mapped static structures in Chimera and/or render movies with the appropriate color mapping shown from 6 points of view X1, X2, Y1, Y2, Z1, and Z2…or alternatively with 2 points of view incorporating a smooth vertical and horizontal roll during playback. These movies can be viewed

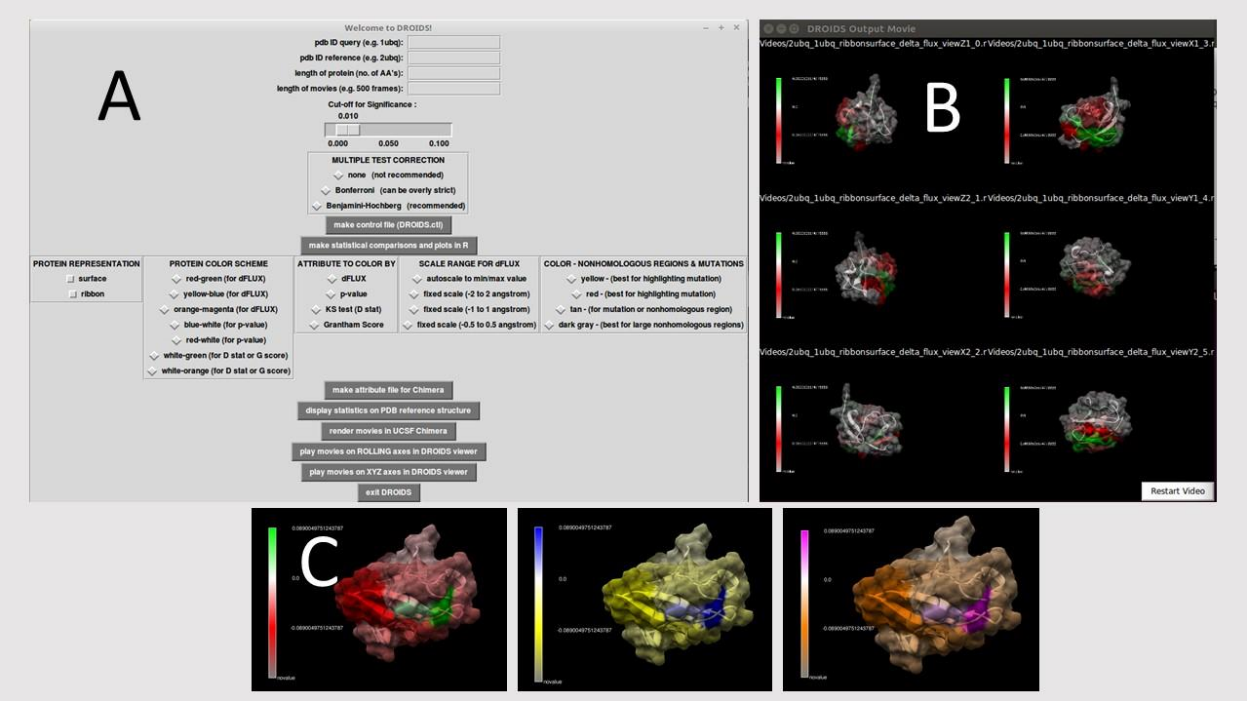

Figure 4. (A) The final DROIDS 1.20 GUI controlling the KS statistics, multiple test correction method and graphics options. (B) A movie viewer showing six points of view (front, back, left, right, top, bottom) is also provided. (C) Color options are quite numerous. Bivariate color options for dFLUX are shown here. Univariate coloring options for p or D values of the KS test are also provided.

simultaneously in concert in the DROIDS movie viewers. While the colors mapped correspond to the overall analysis, the movie dynamics correspond to only the first MD sampling run taken on the reference PDB structure.

Current and future uses for DROIDS 1.20

The potential uses for DROIDS are many. Some ideas we have imagined during its development include the visualization of the functional effects of natural and artificial mutation at the protein sequence level (i.e. amino acid replacement and site-directed mutagenesis). Population variation associated with disease related malfunction (i.e. nsSNV – nonsynonymous single nucleotide variant) might also be analyzed. The functional impacts of post-translational modifications (e.g. disulfide bridging or phosphorylation) and epigenetic modifications (e.g. acetylation and methylation) will also be of considerable interest on computers that can handle larger molecular systems. Functional consequences of natural evolutionary divergences created through the processes of speciation, gene duplication and genetic drift / genomic decay can also be compared. Future releases will include methods of distinguishing selection from drift. The study of functional binding interactions (protein-ligand, protein-DNA and protein-protein) will be possible upon future development of versions of DROIDS that can analyze multi-chain systems. Currently, the code is limited to single chain comparisons. Null comparisons are

also useful. These are when the exact duplicate of the same PDB files are run through DROIDS. Because molecular dynamics can diverge wherever the system does not settle into potential energy wells, a null comparison on a single structure using DROIDS can show users where the MD is potentially failing to replicate reproducible biophysics. Additionally, mutations that disrupt structural autocorrelations may prove interesting as well. We hope to design future editions of DROIDS that are specific to particular areas of molecular evolutionary biology (e.g. chromatin dynamics, transcription factor binding, G protein activation, Heme protein comparison etc.)

Two examples using DROIDS

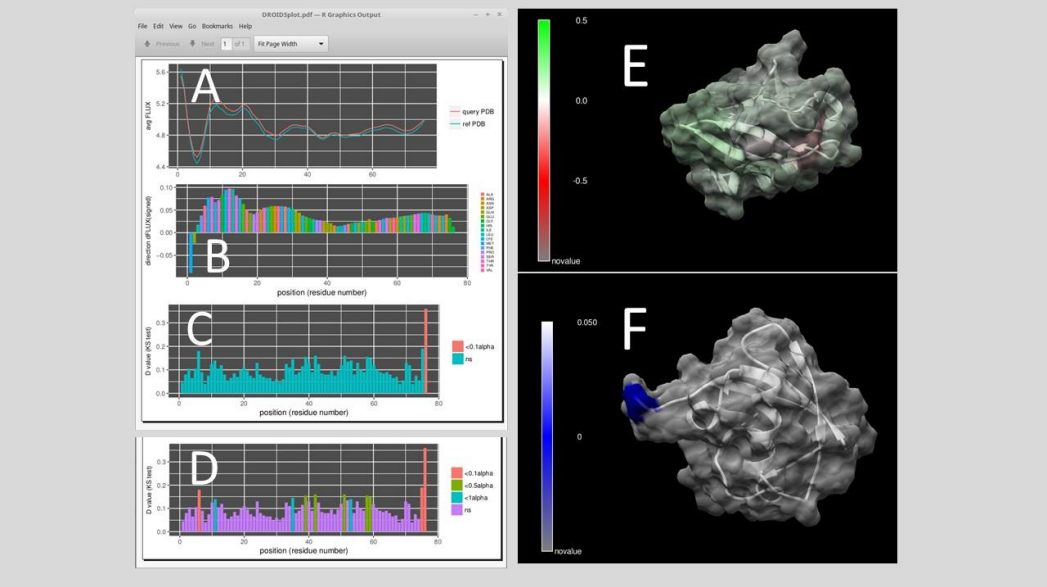

Figure 5. A null comparison of molecular dynamics on a small protein (PDB 1ubq – 1ubq). The profiles in average FLUX as a function of position are nearly identical. (B) The differences (i.e. dFLUX) are colored as a function of amino acid type are (C) almost entirely non-significant except at the terminal end of the protein. (D) Results without correction for false discovery rate are also shown for comparison. (E) dFLUX and (F) p values of the KS tests are shown color mapped to PDB 1ubq.

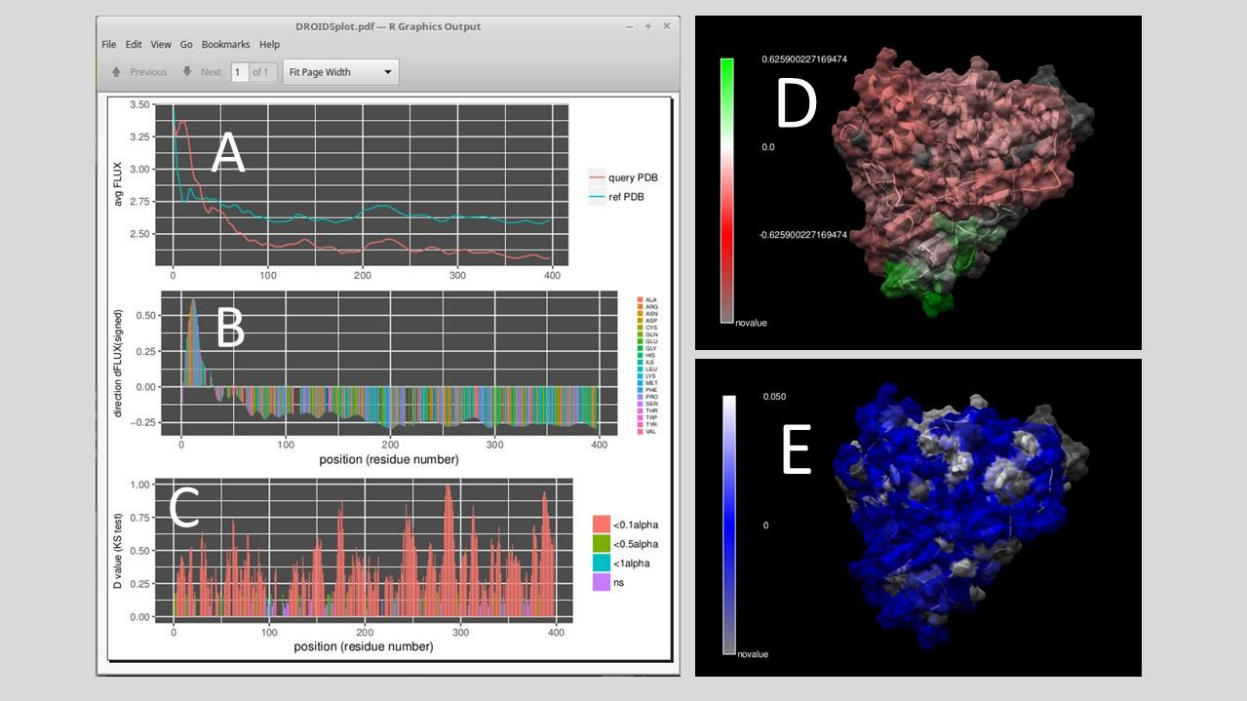

Figure 5. A comparison of molecular dynamics on a thermostable and wildtype p450 cytochrome (PDB 1t2b – 1phd). The profiles in average FLUX as a function of position quite different. (B) The differences (i.e. dFLUX) are colored as a function of amino acid type are (C) almost entirely significant across most of the protein. (E) dFLUX and (F) p values of the KS tests are shown color mapped to PDB 1phd.