Human learning for molecular simulations:
the Collective Variables Dashboard in VMD

Jérôme Hénin[*,1,2], Laura J. S. Lopes[3], and Giacomo Fiorin[4]

[1]Laboratoire de Biochimie Théorique UPR 9080, CNRS, Université de Paris, 75005 Paris, France

[2]Institut de Biologie Physico-Chimique–Fondation Edmond de Rothschild, PSL Research University, 75005 Paris, France

[3]Theoretical Division T-1, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, USA

[4]Theoretical Molecular Biophysics Laboratory, National Heart, Lung and Blood Institute, National Institutes of Health, 10 Center Drive, Bethesda, MD 20814

October 19, 2021

Abstract

The Collective Variables Dashboard is a software tool for real-time, seamless exploration of molecular structures and trajectories in a customizable space of collective variables. The Dashboard arises from the integration of the Collective Variables Module with the visualization software VMD, augmented with a fully discoverable graphical interface offering interactive workflows for the design and analysis of collective variables. Typical use cases include a priori design of collective variables for enhanced sampling and free energy simulations as well as post-mortem analysis of any type of simulation or collection of structures in a collective variable space. A combination of those cases commonly occurs when preliminary simulations, biased or unbiased, reveal that an optimized set of collective variables is necessary to improve sampling in further simulations. Then the Dashboard provides an efficient way to intuitively explore the space of likely collective variables, validate them on existing data, and use the resulting collective variable definitions directly in further biased simulations using the Collective Variables Module. We illustrate the use of the Dashboard on two applications: discovering coordinates to describe ligand unbinding from a protein binding site, and designing volume-based variables to bias the hydration of a transmembrane pore.

[*]Corresponding author: `henin@ibpc.fr`

1   Introduction

Research using biomolecular modeling and simulations nearly always resorts to geometric descriptors for analyzing molecular trajectories, or structures and their differences. Such descriptors are low-dimension functions of atomic coordinates, and are often referred to as *collective variables* (abbreviated to colvars). Collective variables can be used either for post-mortem analysis of sets of structures and simulated trajectories, or to modify simulations on the fly in order to restrain, enhance, or otherwise bias the motion of chosen degrees of freedom. When preparing a biased simulation, these variables are chosen based on previous knowledge. Constructing such a reduced space of relevant coordinates constitutes a dimensionality reduction process. This step may be performed as an automated, data-driven approach, where dimensionality reduction algorithms determine optimal collective variables for a given problem.[1,2] Still, these approaches are a topic of active research, and have not reached the stage where they can be applied as routine, black-box tools. Furthermore, an intuitive interpretation of the reduced representation is always desirable eventually. Some dimensionality reduction methods are explicitly aimed at constructing interpretable coordinates.[3] In most current applications, the dimension-reduction process is knowledge-based rather than data-driven: the practitioner's intuition, guided by available information, is the driving force. Choosing and refining the relevant collective variables requires a large expense of human time and effort. Thus, any workflow using collective variables sooner or later requires an intuitive connection between the space of atomic coordinates and the space of collective variables. These variables are typically analyzed and visualized using combinations of more or less ad hoc analysis software and generic plotting tools.

The new tool we publish here provides an immediate, intuitive connection between existing molecular data and a very broad space of possible collective variables, resulting in more robust reduced descriptions with reduced researcher time and effort. It provides a concrete, visual sense of the abstract mathematical objects that are collective variables. In our hands, this has proven just as useful to scientists in training as to experienced practitioners.

A fundamental everyday tool of molecular biophysicists is molecular graphics software, such as VMD.[4] In addition to graphics, VMD already provides some ways to access geometric measurements on static structures or along trajectories. The simplest form consists in picking pairs, triplets and quadruplets of atoms with the mouse to label distances, angles and dihedral angles respectively. Analysis plugins offer more variables that can be analyzed, notably root-mean-square deviations (RMSD), H-bonds and salt bridges, and more collective properties such as contact maps, radial distribution functions, or projections along normal modes. These represent only a limited subset of the variables that are used for biasing or analyzing simulations.

The Collective Variables Module (Colvars)[5] is a software library integrated with the standard releases of major molecular simulation and analysis software packages, including LAMMPS,[6] NAMD[7] and VMD.[4] Additionally, patched releases of GROMACS[8] have also become available recently. Colvars can be used to

modify the simulated dynamics according to various types of biasing algorithms acting on functions of the atomic Cartesian coordinates; we refer to those functions as collective variables, or *colvars* in short. The Colvars Module implements adaptively biased, free energy simulations such as variants of the Adaptive Biasing Force (ABF)[9–11] and metadynamics.[12,13] The complete space of collective variables can be used for adaptive free energy calculations using eABF,[11] which lifts some technical requirements of ABF.[9,14] Within VMD, the Collective Variables Module functions as an analysis tool, allowing for computation and analysis of colvars through a scripting interface based on the Tcl language used by VMD's command line interface.

The workflow typically begins with exploratory, unbiased simulations. The resulting trajectories can inform the design of biased simulations aimed at more fully exploring regions of interest and sampling those metastable degrees of freedom that are deemed relevant by the scientist. Information gleaned from biased simulations may in turn suggest an improved space of collective variables, resulting in the iterative workflow illustrated by the solid arrows in Figure 1.
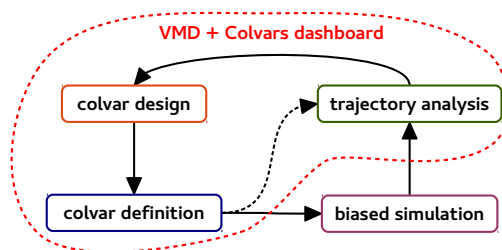

Figure 1: A generic collective variables modeling workflow. Several steps are unified and simplified using the Dashboard, and transfer between analysis and simulation is streamlined thanks to the compatible Colvars Module interface.

This process is limited by two types of obstacles. The fundamental obstacle is the difficulty of defining and discovering the most relevant degrees of freedom in high-dimensional systems; this may require iterations on any existing structural or simulation data. Technical obstacles arise in the various steps necessary to design, implement and calculate collective variables on-the-fly or when post-processing simulations. The Collective Variables Module (*Colvars*) is a broad-purpose toolbox, and writing correct input files making use of all its features requires some time and care. Analyzing trajectories may involve running several pieces of specialized software in succession, either manually, or with the help of in-house scripts. The Colvars Dashboard is a software helper designed to accelerate this process going from data to information, to knowledge. Even In the case of a purely data-driven dimensionality reduction such as PCA, dihedral PCA,[15] FCA[16] or TICA,[17–19] the Colvars Dashboard allows for inspecting the data in the reduced space, to build *a posteriori* a physical intuition about the reduced space.

There are currently graphical interfaces for VMD that interact with the collective variable software PLUMED:[20] Plumed-GUI[21] as a writing aid, and METAGUI[22,23] for clustering and free energy analysis. Both plugins invoke PLUMED as an external executable, passing data from VMD's memory as temporary files; in addition, PLUMED-GUI offers configuration templates and an interface for plotting variables using

the Colvars Module. However, the present tool provides a much broader access to the functionality of the Colvars Module. In particular, it exploits the binary-level integration of Colvars with VMD to provide a fully interactive workflow where variables are updated in real-time and the molecular and collective variable visualizations are tightly linked.

Use cases for the Colvars Dashboard include:

- Defining colvars *a priori* based on a single known structure;

- Exploring different colvars on a set of known structures;

- Analyzing simulated trajectories in colvar space;

- Refining colvars based on simulated trajectories.

Below we describe the design and implementation of the Dashboard, explain its uses through two types of typical tasks (visualizing a given set of variables, and designing new variables). We then illustrate its functionality on two real-world scientific applications: exploring protein-ligand binding and unbinding, and designing and visualizing variables based on volumetric maps.

2 Design and implementation

The functional, user-visible design of the Dashboard is summarized in Table 1.

Task	Technical solution
explore previously defined colvars	load config file
use colvars from Dashboard in future simulations	save config file
rapidly create new colvars	colvar templates
choose among available components	link to list of components in online documentation
rapidly define chosen components	component templates and online documentation
define atoms involved in variables	insert atoms from representations, labeled atoms, or VMD selections
double-check atom selections	show colvar atoms in molecular representation
explore dependence on atomic coordinates	colvar gradient visualization
explore dependence on specific molecular motion	interactively move atoms/molecules and update colvars
explore time-behavior of variables	timeline plots
explore correlation between variables	pairwise scatterplots
visualize specific high-dimensional quantites	rotation display, volume map display
other quantitative analysis	export timeline to external software

Table 1: Summary of useful tasks in a collective variable workflow, and the corresponding technical solutions implemented by the Colvars Dashboard

Internally, the Colvars Dashboard is written in the scripting language Tcl, and calls four programming interfaces:

1. the Collective Variables Module, to create and query colvars;

2. VMD, to drive the display of specific trajectory frames, interact with atom selections and representations, and visualize Cartesian gradients;

3. Tk, to create the graphical user interface;

4. the VMD plugin Multiplot, for interactive plots.

The Collective Variables Module (*Colvars*) is already included in official binary releases of VMD, and thus available to all users on Linux and MacOS platforms since version 1.9.2, and on Windows platforms starting with VMD 1.9.4. The Dashboard graphical plugin is also distributed with VMD 1.9.4 and later (the recommended version as of publication is 1.9.4a54). The latest version can be downloaded at any time from the Colvars public repository on GitHub (`https://github.com/Colvars/colvars/tree/master/vmd/cv_dashboard`), and updated in an existing VMD installation without recompiling.

2.1 Underlying library: the Collective Variables Module, distributed with VMD

Colvars[5] is a portable C++ library, with an interface class `colvarproxy` that allows for writing new interfaces to C++, C, or Fortran programs without modifying the core of Colvars. In MD simulation programs, Colvars is called at every timestep to update variables and biases as needed. In contrast, in VMD, access to Colvars features is enabled *on-demand*, through its rich Tcl scripting interface. This interface, already available since the Colvars versions included in the VMD 1.9.3 release, allows for dynamically allocating the library, creating and deleting variables and biases, parsing configuration files, and computing quantities such as the variables' values and atomic gradients on the fly, using the current atomic coordinates stored by VMD in memory. The Dashboard here introduced leverages this existing scripting interface, and binds the most commonly used functionality to simple graphical objects such as menus and buttons.

2.2 Discoverability and documentation

The Dashboard's graphical interface is designed to be simple and discoverable. Rather than menus, all functions are visible as clearly labeled buttons, making the learning curve quite smooth. Power users may use an extensive collection of keyboard shortcuts, some of which are directly mentioned in the interface, while all are included in the documentation.

The documentation is available online as part of the VMD edition of the Colvars user's guide.[24]

The Configuration Editor window, the use of which requires the most background information, features direct links to the relevant sections of the online documentation. The template configurations provides syntax examples for virtually all features, while the online documentation provides a reference and background explanations.

2.3 Interaction with MD engines

To ensure a smooth workflow, the same Colvars configuration file should be used in sequence in the Dashboard and in MD simulations, without modifications. The compatible MD engines are all those with functional interfaces with Colvars: LAMMPS,[6] NAMD,[7] and our public, patched version of GROMACS.[8] Most of these programs use different unit systems from each other: in particular, LAMMPS is used for a broad variety of application domains and offers a choice of multiple unit systems that can be set by the user. To enable interoperability with all MD engines, the Colvars/VMD interface performs unit conversion between VMD's internal units and the current set of units for Colvars. The unit system can be changed within the graphical interface. A "units" flag is added to the Colvars configuration to avoid accidental use of inconsistent units: running Colvars with a specific MD engine will trigger an error condition if the configuration specifies units that are inconsistent with those of the engine.

3 Task 1: Visualizing a given set of variables

The main window 2 features a two-column table listing the variables on the left, and their values for the current timestep on the right. Vector variables such as vector distances or orientation quaternions can be expanded to select their scalar components individually for plotting. By default, values are updated in real-time to track the current trajectory frame displayed by VMD, however this can be disabled especially for computationally expensive sets of variables that are not suitable for real-time display.
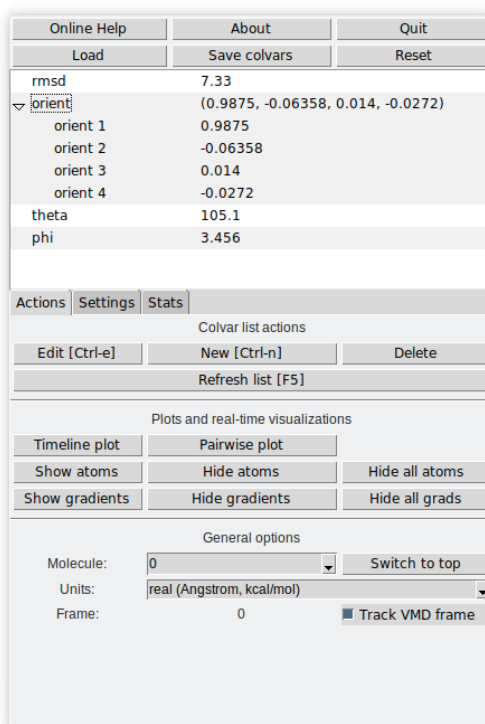

Figure 2: The Collective Variables Dashboard main window.

6

3.1 Loading and saving configuration files

Key to integration of the interactive colvar modeling process into the simulation workflow is the ability to read and write the same configuration files that would be used in biased simulations in any of the supported MD engines. The selection of colvars that has been refined interactively can be saved at any time. The Dashboard only handles collective variables themselves, not biasing methods. therefore, the configuration of biases should be kept in a separate file and loaded separately in MD simulations engines.

3.2 Visualizing trajectories simultaneously in VMD's molecular representations and in colvar space
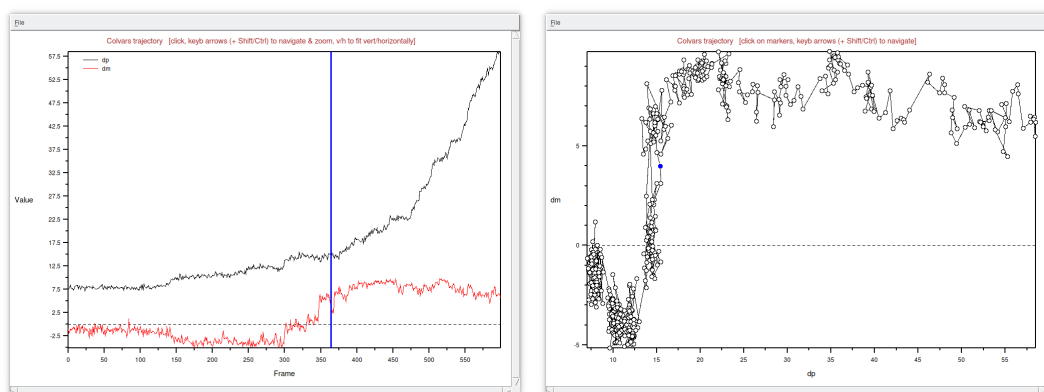

Figure 3: Plots of collective variable values along a loaded trajectory. Left: Timeline plot. Right: Pairwise plot. The vertical blue line in the timeline and blue dot in the pairwise plot indicate the current frame in VMD.

The timeline view provides a plot of the values of a selection of variables as a function of time (Figure 3). A vertical time cursor points to the current frame of the trajectory being displayed, and moves when the trajectory is animated. Conversely, clicking on any point in the timeline window brings VMD's display to that frame in the trajectory. Left and right arrow keys on the keyboard navigate in the trajectory, while up and down keys provide a zoom in/zoom out functionality to access detailed features of long trajectories.

The pairwise scatter plot can be created for pairs of scalar colvars (or scalar components of vector variables), and is useful for exploring correlation between variables (see Figure 9). As in the timeline view, trajectory animation is visualized, with a colored dot representing the current frame. A click on a dot brings the VMD representation to the frame in question.

3.3 Exploring the dependence of colvars on atomic coordinates

Atom coordinates can be modified interactively using VMD tools (translate and rotate atoms, residues, and molecules using a mouse or another device). Then the table of colvars can be updated almost instantly by pressing F5, allowing for a near-real-time exploration of changes in collective variables in response to

changes in structure.

In addition, the Dashboard features a real-time visualization of the gradient of an arbitrary scalar variable with respect to Cartesian coordinates (Figure 4). In this visualization, the gradient of the variable with respect to the position of each atom is represented by an arrow starting from the atom in question. Like the colvar values, the gradients are computed on the fly, and can be updated near-instantly after a change in coordinates. This is particularly useful for variables whose mathematical expression is too complex to build a clear intuition, such as dihedral principal components[15] (Figure 4), or distance to bound configuration[25] (Figure 11).
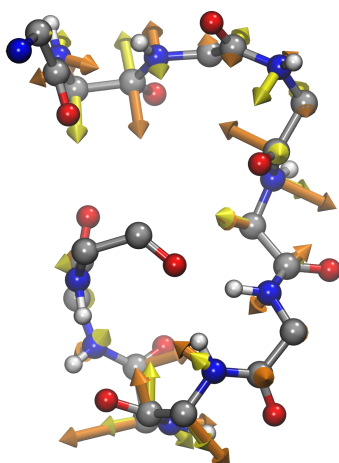

Figure 4: VMD molecular representation of the backbone of deca-alanine, featuring the atomic gradients of its first two dihedral principal components,[15] as defined in reference.[26] Gradients of the the first and second mode are represented by yellow and orange arrows, respectively. The arrow length scale is arbitrary but identical for both coordinates, allowing for comparing the gradient magnitudes.

3.4 Dedicated visualizations for specific quantities

Colvars offers a special type of variable to represent rotation operators, encoded as unit quaternions. In particular, the `orientation` component describes the collective rotation of a group of atoms with respect to reference coordinates, computed based on a roto-translational least squares fit.[5] This is particularly useful to capture the collective rotation of a flexible object in a robust way. Rotations have three degrees of freedom (e.g. two defining the axis and one defining the angle of rotation), and are difficult to visualize in an intuitive way. The Dashboard offers a specific representation for such variables, illustrated in Figure 5. An arrow represents the axis of rotation, oriented so that the rotation angle is between 0 and 180 degrees in the direct (counterclockwise) sense. The angle itself is represented by two segments protruding from the axis.

The Dashboard also facilitates the visualization of volume maps associated with existing colvars: see Section 5.2 for details.
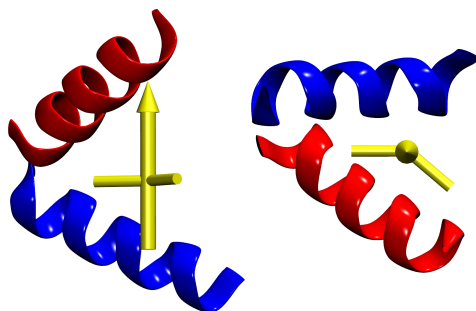
Figure 5: Graphical representation of a rotation operator characterizing the orientation of a molecule with respect to a reference, viewed orthogonal to (left) and along (right) the axis of rotation. Here the molecule of interest (red) and the reference (blue) are two copies of a 15-alanine helical peptide. The rotation operator is represented by the yellow arrow, oriented along the axis of direct rotation, and the angle between the two side "arms" is the angle of rotation.

4 Task 2: Designing new collective variables

4.1 Editing colvar configurations

The Dashboard includes a Collective Variable Editor dialog, which presents the text of the configuration string, as well as a number of helper features that make creating and modifying variables much faster and less error-prone. The editor contains direct links to relevant sections of the online documentation. To soften the learning curve, most of the features can be inserted as templates which require only limited modifications. In addition, possible syntax errors are handled gracefully: while the configuration of an existing variable (or several variables) is being edited, its previous working configuration is backed up. If the user submits an edited configuration string and Colvars returns an error, the complete error message is displayed, and the editor window is shown again, providing a chance to fix the error. If the window is closed before any changes can be applied successfully, the saved, functioning version of the variable(s) is restored.

4.2 Inserting atom selections

A significant part of defining collective variables is to specify which atoms are involved in the various groups used to calculate the variable. The Dashboard provides several helper features to create atom selections:

1. From a VMD atom selection text. This helps leverage the very powerful atom selection language of VMD. Furthermore, atom selections created this way are dynamic: the selection text is embedded in the Colvars configuration string as a special comment, and recomputed every time the configuration is parsed, possibly adapting to changes in coordinates, or topology if the same configuration is applied to different molecules.

2. From a VMD representation, selected via a dropdown menu. This lets the user immediately transpose

atom groups being visualized into collective variables.

3. From "picked atoms" selected using the mouse in VMD's graphical window.

The last helper is a file picker dialog, which allows for inserting the paths of files containing reference coordinates or atom group definitions, together with the relevant keywords.
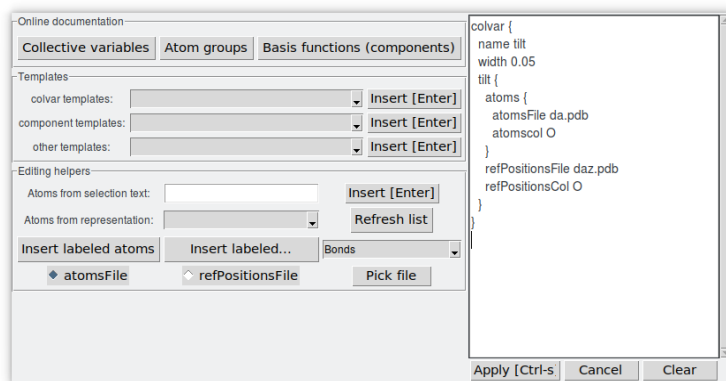

Figure 6: Colvar configuration editor window.

Atom groups used by existing variables can be visualized by selecting them and clicking "Show Atoms" in the main window. This creates VMD representations showing atoms as spacefill, with distinct colors for each atom group. This is useful to check that the designated atoms correspond to those intended, preventing issues such as off-by-one errors due to conflicting atom numbering schemes between software. Note this is the reverse operation to creating an atom group directly from a representation.

4.3 Inserting templates

The editor interface allows for inserting templates for all types of basis functions, and for colvars with specific options whose syntax might be difficult to recall. The templates contain placeholders for key parameters, and comments that function as a first level of inline documentation. More in-depth documentation is available by clicking the documentation links.

4.4 Inserting geometric observables from VMD

VMD lets users select and measure distances (*Bonds*), angles, and dihedral angles between particles by selecting them with the mouse in the graphical window. These geometric quantities can be directly inserted as collective variable component definitions.

4.5 Setting up biased MD simulations

Typically, a project will start with unbiased simulations for an agnostic, local exploration of configuration space. The Dashboard greatly reduces the time needed to test new ideas for collective variables based on

any such trajectories, biased or unbiased, to use in further biased simulations. Although major MD engines support "reruns", analysis of colvars in VMD is much quicker, and the new variables are evaluated frame by frame in real time. This is even more relevant when analyzing simulations made with one MD engine but planning simulations with another one to take advantage of unique features in each package. The Colvars Module allows this transferability, and the Dashboard facilitates it. In particular, changes in unit systems between different MD engines are handled gracefully and safely.

5 Example applications

5.1 Exploring ligand binding and unbinding

5.1.1 Monitoring ligand unbinding simulations

Quantitative studies of ligand unbinding kinetics raise major sampling challenges, which can be alleviated through the use of a number of enhanced sampling methods. Many of those require a precise definition of the bound state as well as a suitable coordinate to track the ligand's progress. In those tasks, the use of a tool like the Dashboard enables the user to test multiple colvar candidates in order to find what best suits the problem. The example discussed in this section consists in the unbinding of a pharmacological ligand from the N-terminal domain of the heat shock protein 90 (Hsp90) chaperone (Figure 7). The system was first explored using Adiabatic Bias MD (ABMD)[27] (implemented in Colvars/NAMD), which generated a family of possible exit trajectories under a mild non-equilibrium bias. Then an unbiased sample of exit trajectories as obtained using the Adaptive Multilevel Splitting (AMS) method.[28] The Colvars Dashboard was used in the analysis of trajectories generated with both methods.

The analysis of the ABMD trajectories using VMD reveal that the ligand detaches from the protein first by its aromatic ring, followed by its bicycle structure (Figure 7). This suggests the distances from those two sections of the ligand should be considered separately. Figure 8 shows the Dashboard session used in the exploration of new collective variables based on that idea. The time plot confirms that the aromatic ring is the first to disconnect from the protein, but also shows that this represents an intermediate state, as the ligand stays a while bound only through its bicycle.

Another result from the analysis of those trajectories using Colvars Dashboard is that only one component of the distance vector is necessary to describe the ligand's exit. Figure 9 shows the axis used in this new coordinate, along the direction of the cavity opening. The pairwise plot between the distance and the projection onto the plane shows a poor correlation, but the pairwise plot with the projection onto the axis exhibits a high correlation, suggesting this new variable as a suitable progress coordinate. Furthermore, compared with the isotropic distance, it provides a better resolution between states where the ligand is at a small distance from the protein and escaping along the axis leading away from the site, and those where the ligand is interacting nonspecifically with the protein surface outside the cavity.

The two trajectories generated with the AMS method exhibit a long residence time within two distance intervals distinct from the initial bound state, suggesting the existence of two additional metastable states.
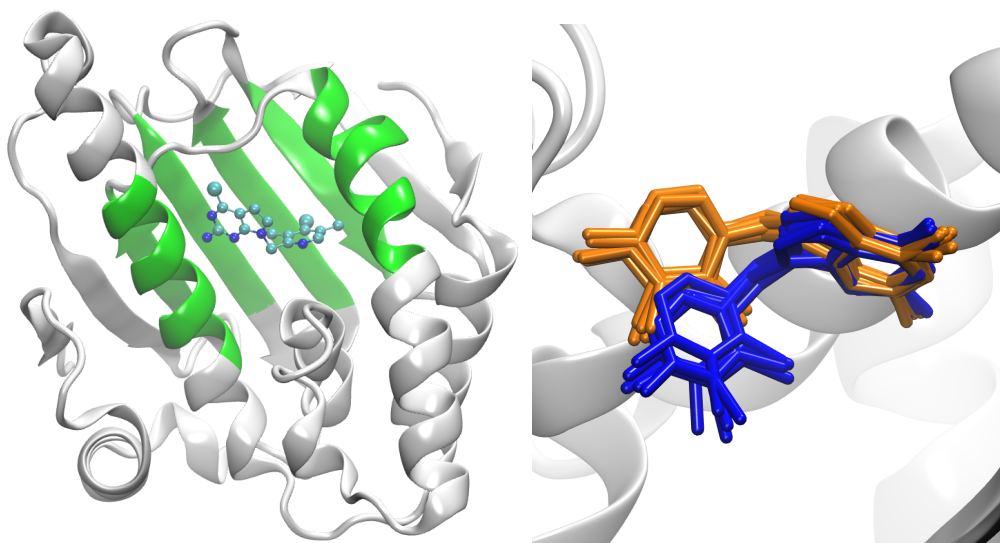
Figure 7: Left) The Hsp90/ligand system. The ligand is represented as ball-and-stick, and the protein as cartoon, with residues chosen to represent the binding site colored green. Right) In a zoomed view of the binding site, the two bound states are represented by a collection of configurations from simulations, shown as sticks: the main bound state is colored blue, and the secondary one is colored orange. The latter was discovered by analyzing ABMD trajectories.
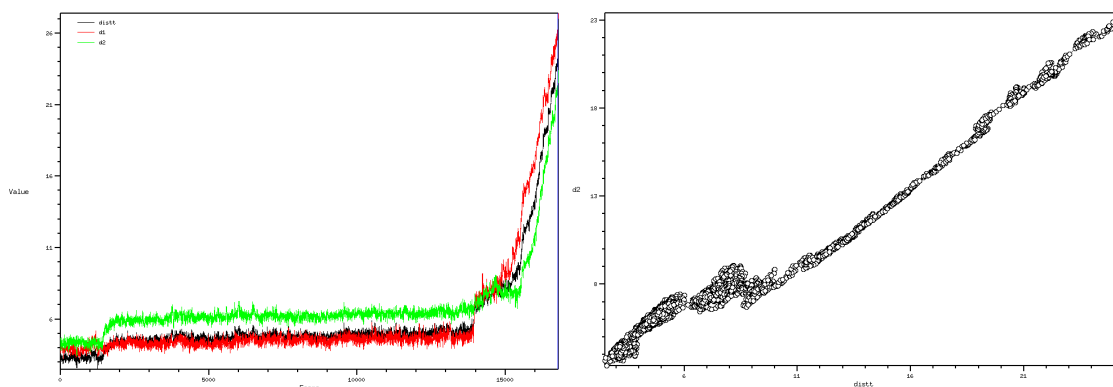

Figure 8: Colvars visualization of ABMD escape trajectories using the Dashboard. Left: Timeline of the center of mass distance to the protein cavity (black), the distance of the bicycle (red) and the distance of the aromatic ring (green). Right: Pairwise plot that correlates the global site-ligand distance `distt` with the distance from the aromatic ring `d2`

Because those intervals did not coincide but did overlap, it was necessary to find a new coordinate that could separate them, to distinguish them within the AMS approach.

A rapid visual inspection of the trajectories suggested that the larger, bicyclic moiety of the ligand experiences a rotation between the two modes (Figure 10). The Dashboard allowed for quickly testing rotation around different axes to separate those modes. Figure 10 show plots for the global rotation angle from the reference (`orientationAngle` component), and angles of rotation around selected axes (`spinAngle`
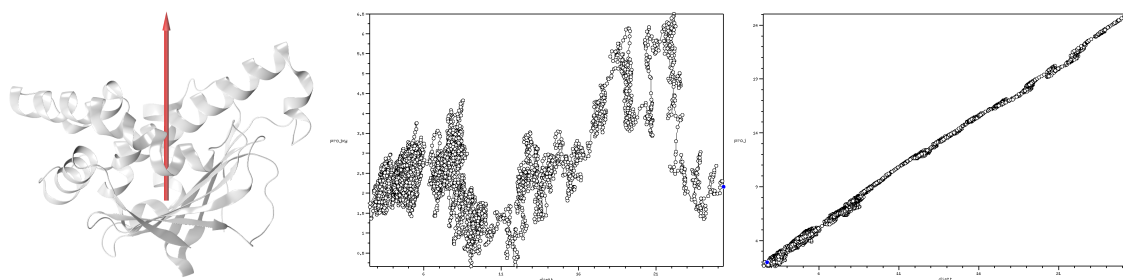
12

Figure 9: The axis (in red) used in the new explored coordinate. Both pairwise plots show the correlation between the distance to the cavity and its projection onto, respectively, the plane parallel to the cavity opening (left), and the axis perpendicular to it (right). The latter exhibits a high correlation with the isotropic distance in this trajectory.
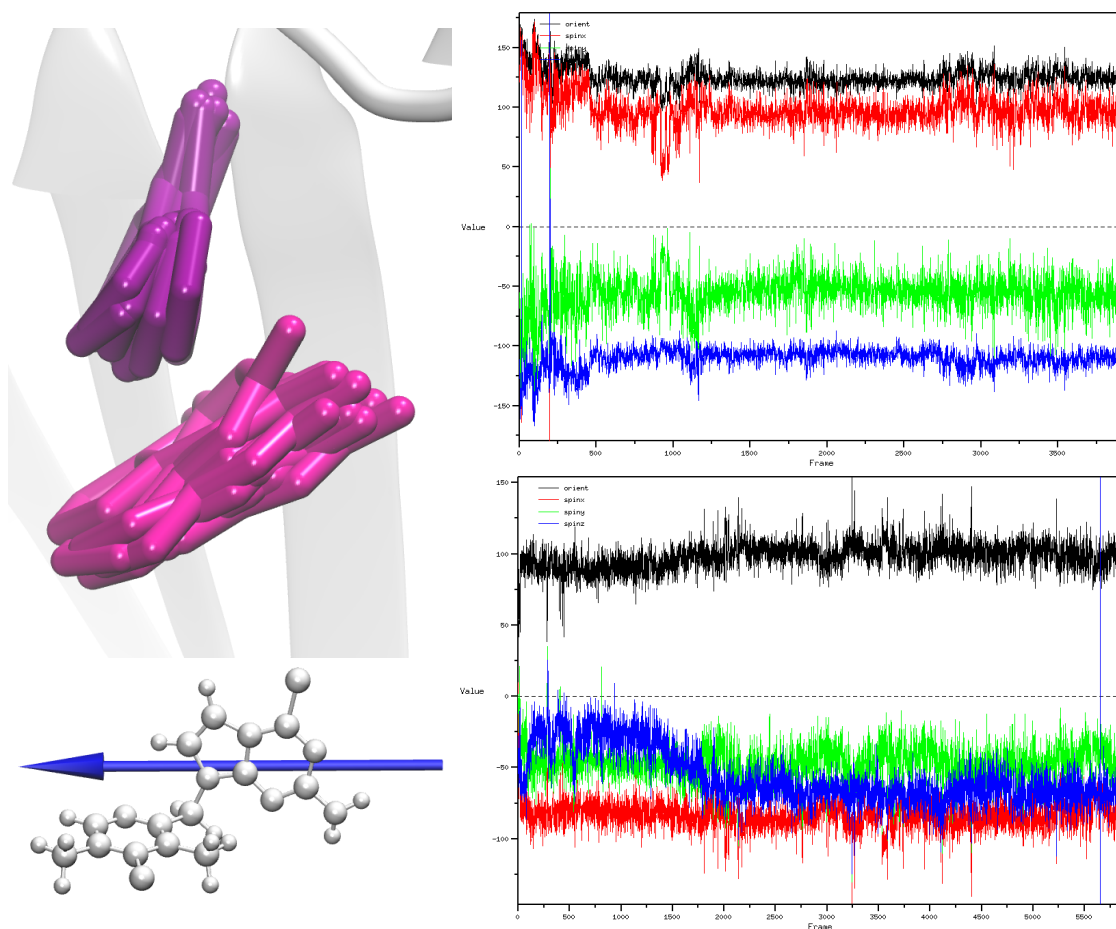

Figure 10: Left) Positions of the ligand's bicyclic moiety for the two AMS trajectories. The two modes differ by a rotation. At the bottom the axis used for the coordinate candidate (*spinx*). Right) Timeline plots of 4 candidate coordinates along trajectories sampling each of the two states (upper and lower panel, respectively). The angle of rotation around the x axis (*spinx*, red curve) proved a solid coordinate to separate the two states.

component).[5] The time plot of the different trajectories shows that the rotation angle around the x axis (in blue in Figure 9) was successful in separating the states. These orientations of the ligand were computed relative to the protein, using the moving frame of reference formalism inherent to Colvars.[5]

In this example the use of Colvars Dashboard was key in the exploration of unknown metastable states, the design of a better suitable progress coordinate, and the separation of metastable states. In the latter, the comparison between different coordinates was critical, as the relevant axis of rotation could not be determined by eye. Finally, the Colvars Dashboard was able to aid in the setting of new unbinding simulations, both in the progress coordinate and in the coordinates used to define different bound states.

5.1.2 DBC coordinate for ligand binding

Computational prediction of absolute binding affinities can also require careful design to ensure convergence, including a potentially complex ensemble of binding restraints connected to the definition of the binding site.[29] The Streamlined Alchemical Free Energy Perturbation (SAFEP) approach[25] offers a simpler protocol based on flat-bottom restraints on a single, scalar binding coordinate named distance-to-bound-configuration (DBC). DBC is a ligand RMSD with respect to an ideal bound pose, computed in a moving frame of reference tied to the binding site.[25,30] DBC captures the ligands conformational changes as well as its translational and rotational motion with respect to the binding site. Crucially, DBC is orthogonal to all internal degrees of freedom or the receptor, so that restraint forces on DBC do not bias the receptor's internal dynamics.

As the DBC coordinate aggregates many degrees of freedom, it is helpful to make its meaning more intuitive by visualizing its gradient for different ligand configurations. Close to the reference bound pose, DBC is mostly sensitive to the ligand conformation and orientation (as in Figure 11), while at larger values, it reflects mainly global translation away from the site.

Multimodal binding requires particular care to be correctly described, especially in quantitative affinity predictions.

$$e^{-\beta \Delta G} = \sum_i e^{-\beta \Delta G_i} \ , \tag{1}$$

where ΔG is the global effective binding free energy, ΔG_i is the free energy of binding to individual mode i, and $\beta = 1/k_B T$. Different modes in the DBC distribution (or those of other descriptors) can easily be related to different binding modes using the timeline view of the Dashboard.

5.2 Volumetric maps and Multi-Map variables

5.2.1 Support for volumetric maps by the Colvars module and the Dashboard

When using enhanced-sampling MD in biophysical applications, two frequent requirements are the ability to tackle complex changes in macromolecular shape, and the ease of implementation of a suitable CV. One approach aimed at attaining both is defining a CV as a function of a *volumetric map*, expressed on a three-dimensional grid of Cartesian coordinates. Additionally, the use of multiple maps to represent the distinct
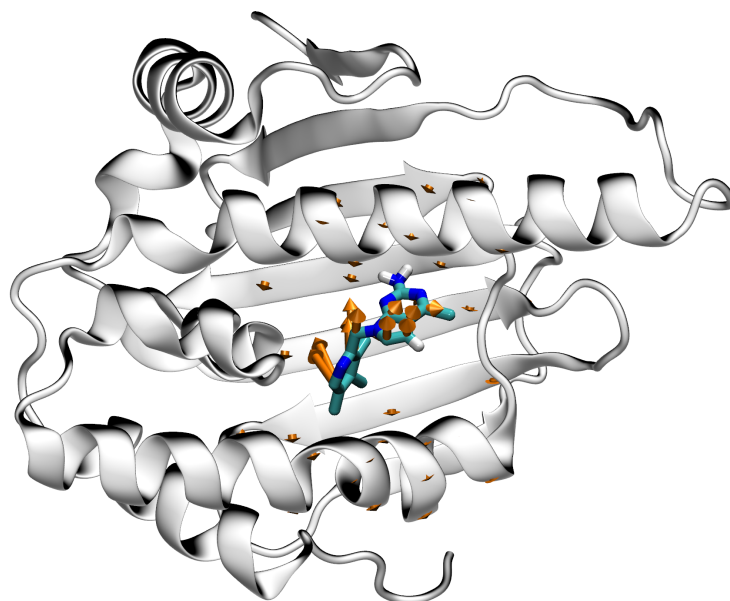
Figure 11: Gradients of the Distance-to-Bound-Configuration (DBC) coordinate visualized on a bound pose of Hsp90 and a pharmacological ligand. The arrows represent gradient components on atoms of the ligand and those protein atoms involved in the binding site definition. The gradient components on ligand atoms represent roto-translational as well as conformational degrees of freedom, whereas the components on protein atoms correspond to the purely roto-translational, rigid-body motion obtained by least-squares fit.[5,25]

relevant states of one macromolecule (e.g. a protein) or many molecules (e.g. a lipid membrane) allows for reliable thermodynamic sampling, a capability provided by the Multi-Map method.[31]

The Colvars Dashboard is uniquely useful in assisting with the preparation and the analysis of volumetric map and Multi-Map collective variables. VMD already provides functionality to analyze and manipulate atomic structures alongside volumetric maps, a feature that has been exploited previously in the framework of the MDFF structural refinement protocol.[32,33] Recently, the Colvars module has been extended with the ability to access directly the volumetric maps loaded in VMD's memory, and compute CVs based on volumetric maps, such as "Multi-Map" variables.[31] The Colvars Dashboard leverages this capability to assist the two Tasks listed above, allowing to compute volumetric map-based CVs within VMD. This can be particularly advantageous when the maps themselves are being prepared in VMD, such as the case of Multi-Map CVs based on protein density maps, as exemplified in ref.[34]

Additionally, the Dashboard's graphical user interface allows for one-click selection and visualization of the maps themselves, consistent with how groups of atoms and gradient vectors are also selected and visualized (see Table 1). Figure 12 shows molecular renders including graphical representations of the volumetric maps generated through the Dashboard. This is achieved in a convenient manner by querying VMD for the minimum and maximum values of each map, and using the mid-point between them to define a reasonable contour level (of course, such choice may be customized per user preference).

5.2.2 Example application: wetting/dewetting of a region confined in a protein's structure

Beyond inducing shape changes in membranes[31] or proteins,[34] a useful biological application of volumetric map-based CVs is enhancing the sampling of different wetting states in a confined protein cavity. A simplified example of this was already shown in ref.,[31] using a single-wall carbon nanotube in water as a minimalistic system. Here, to illustrate the use of map-based CVs on a biologically significant system, we consider the transmembrane pore of the M2 proton channel of the influenza virus. The pore of this channel contains several clusters of water molecules, many of which resolved crystallographically,[35] in dynamic exchange with each other and with the outer water buffer. Using a previously equilibrated snapshot of the protein embedded in a lipid bilayer[36] and following an additional 50 ns of equilibrium MD simulation at 300 K and 1 atm, the spatial distribution of water molecules in the pore region was examined a different levels of resolution.

Three different representations of the water density in the pore were used to generate map-based CVs for enhanced sampling. First, the distribution of water oxygen atoms in a cylindrical region of 6 Å radius and 40 Å length was computed using VMD's `volmap occupancy` command (Fig. 12A). Second, the same distribution was convoluted to a Gaussian kernel with half-width $\sigma \simeq 1.5$ Å, using the `volmap density` command (Fig. 12B). Third, a smoothed step-function was used to define a volumetric map equaling 1 within the cylindrical region described above and zero outside of it; a smooth transition was modeled as the right branch of a Gaussian function (Fig. 12C):

$$\phi(r) = \begin{cases} 1 & \text{if } r \leq r_0 \\ \exp\left(-\frac{(r-r_0)^2}{\sigma_r^2}\right) & \text{if } r > r_0 \end{cases} \tag{2}$$

where r is the radial distance from the pore axis, $r_0 = 6$ Å, $\sigma_r = 1$ Å and similarly for the axial coordinate z with $\pm z_0 = 19$ Å and $\sigma_z = 1$ Å.

All three maps were recorded on three-dimensional grids with spacing of 1 Å. In the remainder, any small discrepancies between volumetric map values computed in VMD and NAMD due to different interpolation methods (linear vs. cubic splines) are ignored; for additional details on this issue, consult the Multi-Map tutorial.[37]

The maps were loaded into VMD, and each was used to define a CV based on the `mapTotal` Colvars function.[31] The number of water molecules in the cylinder, as well as the values of the three variables computed by Colvars/VMD are plotted in Fig. 12D-F, along with their block-averages over 1 ns and 10 ns segments.

Steered-MD simulations were carried out using a moving harmonic restraint with a force constant of approximately $(1,000 \text{ kcal/mol})/\langle\xi\rangle_{\text{eq}}^2$, where $\langle\xi\rangle_{\text{eq}}$ is the mean value of ξ in each equilibrium MD simulation (Fig. 12D-F); the center of the restraint was initialized at $\xi_0 = \langle\xi\rangle_{\text{eq}}$ and moved toward $\xi_0 = 0$ in 50 ns. The trajectories of the values of ξ (Fig. 13A-C) show that all three definitions are suitable for artificially changing the number of water molecules in the transmembrane pore, with the corresponding water molecule count n decrease from $\sim$60 to about $\sim$20. The latter value is approximately the number of
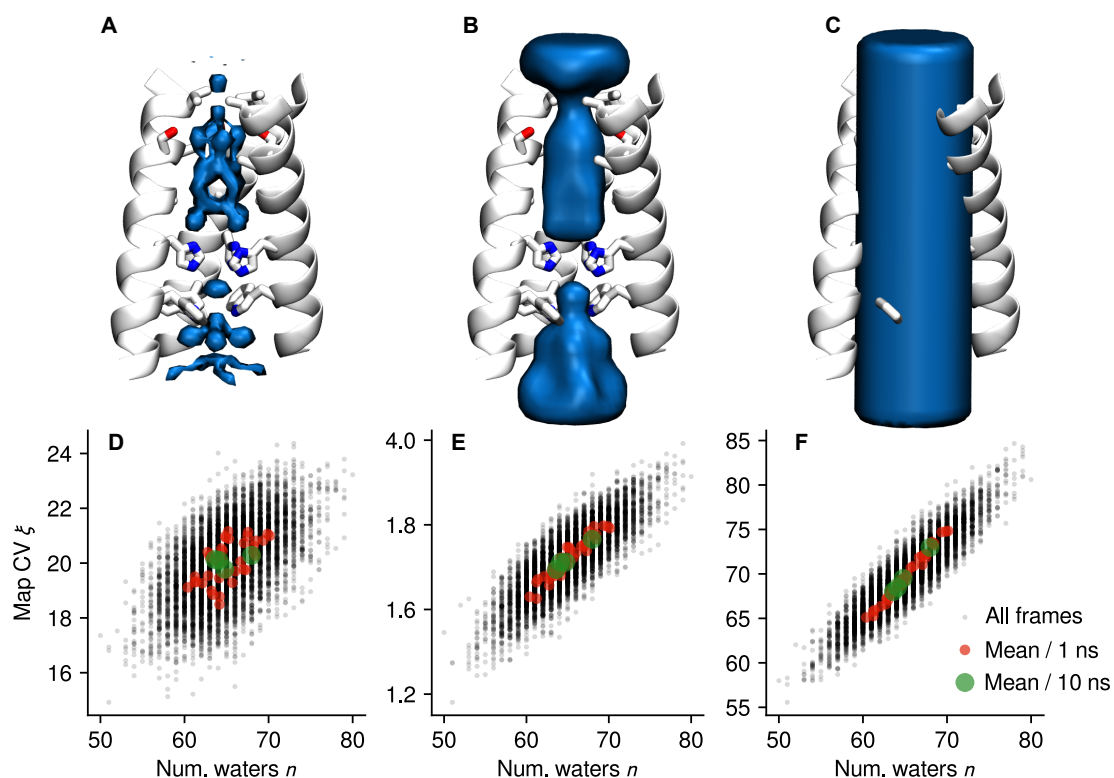
Figure 12: Shown are the M2 transmembrane channel (shown as ribbons) as well as three volumetric maps containing respectively: **(A)** the unsmoothed histogram of oxygen atoms from equilibrium MD simulation, **(B)** the smoothed histogram of the same, and **(C)** a smoothed step function defined on a cylindrical region; each contour is drawn at 50% of the map's maximum value. Also shown in panels **(D-F)** are the values of the CVs based on the same volumetric maps, computed in VMD using the Colvars Dashboard and plotted as a function of the number of water molecules in the same selection.

waters at the two interfaces between the pore and the outer water buffer, indicating that full dewetting of the transmembrane pore was achieved in all three simulations.

Nonetheless, the degree to which each map CV ξ may be used as a proxy for the exact number of waters varies with respect to how smooth the map is. This is particularly true when considering the relatively small equilibrium fluctuations (Fig. 12D-F). For example, using the exact histogram (`volmap occupancy`) leads to relatively poor correlation between the map CV, ξ, and the number of waters, n (Fig. 12D). Using the smoothed density map (`volmap density`) largely improves correlation, both at the equilibrium distribution (Fig. 12E) and under the moving restraint (Fig. 13B). Lastly, the most accurate match between ξ and n is achieved with a cylindrical map, which implements the same selection criteria making up for n following the same cylindrical shape.

The above behavior is attributable to the spatial location of the gradients of each map, which in turn determine the atomic forces. As long as regions with lower water density (e.g. at the pore's boundaries) are geometrically accessible to water molecules, forces aimed at decreasing the corresponding ξ may lead
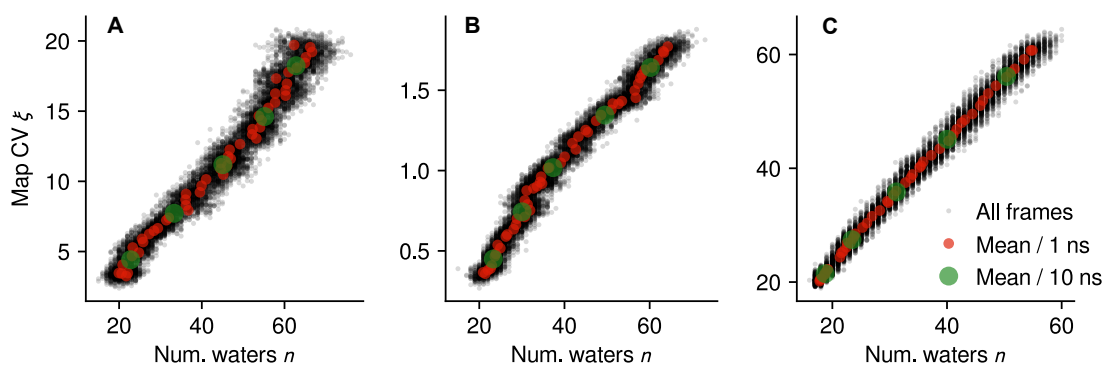
Figure 13: Values of volumetric-map CVs during 50 ns-long steered-MD simulations, with the restraint center initialized near the initial value of each variable and directed toward zero. Similarly to Fig. 12, the CV defined from the occupancy histogram **(A)**, the smoothed density map **(B)** and the cylindrical step function **(C)** are shown.

to movement of water molecules along the membrane plane. On the other hand, if the map decays to zero but far outside of the pore's boundaries (e.g. near the lateral face of the cylinder), forces are only applied to atoms positioned near the two interfaces with the outer buffer (e.g. the two end faces of the cylinder), and only the latter achieves transport inside or outside of the pore.

This example illustrates two important facts:

1. The Colvars Dashboard is useful at speeding up the process of defining a suitable CV based on volumetric maps. In the example shown, best agreement between the CV ξ and the size of the VMD selection n were achieved by using a smoothed version of the same step function that defines the selection itself. Alternatively, the next best choice was provided by a smoothed density map computed from equilibrium MD.

2. Through analysis of equilibrium and non-equilibrium trajectories using the Colvars Dashboard, it becomes easier to define statistically the relationship between the CV ξ and another quantity n, which is in turn more intuitive to understand but at the same time cannot be used as a CV because it is intrinsically discontinuous. That same relationship may be used later to express PMFs computed along ξ as a function of the second more intuitive parameter. This is particularly useful when dealing with *Multi-Map* variables, where the mean values of a relatively complex CV ξ can be expressed in terms of a more physically intuitive parameter.[31]

6 Conclusion

The Colvars Dashboard of VMD constitutes a new type of interface to the versatile Collective Variables Module, making it more useful for the purpose of fostering intuition, while easing its learning curve. The interface is designed to be both accessible to newcomers, and very effective for expert users. It facilitates the

definition of new variables from the wide repertoire afforded by the Module, and their visualization, tightly integrated with the molecular graphics of VMD. The Dashboard also seamlessly integrates with biased simulations using any of the molecular simulation software interfaced with the Collective Variables Module. We expect that the Dashboard will help turn collective variable analysis from a somewhat specialized computational method to an accessible tool for understanding any simulated molecular trajectory.

7 Acknowledgments

References

[1] Florian Sittel and Gerhard Stock. Perspective: Identification of collective variables and metastable states of protein dynamics. *The Journal of Chemical Physics*, 149(15):150901, oct 2018.

[2] Gareth A. Tribello and Piero Gasparotto. Using dimensionality reduction to analyze protein trajectories. *Frontiers in Molecular Biosciences*, 6, jun 2019.

[3] Robert T. McGibbon, Brooke E. Husic, and Vijay S. Pande. Identification of simple reaction coordinates from complex dynamics. *The Journal of Chemical Physics*, 146(4):044109, jan 2017.

[4] W. Humphrey, A. Dalke, and K. Schulten. Vmd: visual molecular dynamics. *J. Mol. Graph.*, 14(1):33–8, 27–8, 1996.

[5] Giacomo Fiorin, Michael L. Klein, and Jérôme Hénin. Using collective variables to drive molecular dynamics simulations. *Mol. Phys.*, 111(22-23):3345–3362, 2013.

[6] Aidan P. Thompson, H. Metin Aktulga, Richard Berger, Dan S. Bolintineanu, W. Michael Brown, Paul S. Crozier, Pieter J. in 't Veld, Axel Kohlmeyer, Stan G. Moore, Trung Dac Nguyen, Ray Shan, Mark Stevens, Julien Tranchida, Christian Trott, and Steven J. Plimpton. Lammps - a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales. *Computer Physics Communications*, page 108171, 2021.

[7] James Phillips, David Hardy, Julio Maia, John Stone, Joao Ribeiro, Rafael Bernardi, Ronak Buch, Giacomo Fiorin, Jérôme Hénin, Wei Jiang, Ryan McGreevy, Marcelo Cardoso dos Reis Melo, Brian Radak, Robert Skeel, Abhishek Singharoy, Yi Wang, Benoît Roux, Aleksei Aksimentiev, Zan Luthey-Schulten, Laxmikant Kale, Klaus Schulten, Christophe Chipot, and Emad Tajkhorshid. Scalable molecular dynamics on CPU and GPU architectures with NAMD. *The Journal of Chemical Physics*, 153:044130, 2020.

[8] Mark James Abraham, Teemu Murtola, Roland Schulz, Szilárd Páll, Jeremy C Smith, Berk Hess, and Erik Lindahl. GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers. *SoftwareX*, 1-2:19–25, sep 2015.

[9] E. Darve and A. Pohorille. Calculating free energies using average force. *J. Chem. Phys.*, 115:9169–9183, 2001.

[10] J. Hénin, G. Fiorin, C. Chipot, and M. L. Klein. Exploring multidimensional free energy landscapes using time-dependent biases on collective variables. *J. Chem. Theory Comput.*, 6(1):35–47, 2010.

[11] Adrien Lesage, Tony Lelièvre, Gabriel Stoltz, and Jérôme Hénin. Smoothed biasing forces yield unbiased free energies with the extended-system adaptive biasing force method. *J. Phys. Chem. B*, 121(15):3676–3685, 2017.

[12] A. Laio and M. Parrinello. Escaping free energy minima. *Proc. Natl. Acad. Sci. U.S.A.*, 99:12562–12565, 2002.

[13] Alessandro Barducci, Giovanni Bussi, and Michele Parrinello. Well-tempered metadynamics: a smoothly converging and tunable free-energy method. *Phys. Rev. Lett.*, 100(2):020603, 2008.

[14] Jeffrey Comer, James C. Gumbart, Jérôme Hénin, Tony Lelièvre, Andrew Pohorille, and Christophe Chipot. The adaptive biasing force method: everything you always wanted to know but were afraid to ask. *J. Phys. Chem. B*, 119(3):1129–1151, 2015.

[15] Alexandros Altis, Phuong H. Nguyen, Rainer Hegger, and Gerhard Stock. Dihedral angle principal component analysis of molecular dynamics simulations. *J. Chem. Phys.*, 126(24):244111, 2007.

[16] Oliver F. Lange and Helmut Grubmüller. Full correlation analysis of conformational protein dynamics. *Proteins: Structure, Function, and Bioinformatics*, 70(4):1294–1312, sep 2008.

[17] L. Molgedey and H. G. Schuster. Separation of a mixture of independent signals using time delayed correlations. *Physical Review Letters*, 72(23):3634–3637, jun 1994.

[18] Guillermo Pérez-Hernández, Fabian Paul, Toni Giorgino, Gianni De Fabritiis, and Frank Noé. Identification of slow molecular order parameters for markov model construction. *The Journal of Chemical Physics*, 139(1):015102, jul 2013.

[19] Christian R. Schwantes and Vijay S. Pande. Improvements in markov state model construction reveal many non-native interactions in the folding of NTL9. *Journal of Chemical Theory and Computation*, 9(4):2000–2009, mar 2013.

[20] Gareth A. Tribello, Massimiliano Bonomi, Davide Branduardi, Carlo Camilloni, and Giovanni Bussi. PLUMED 2: New feathers for an old bird. *Computer Physics Communications*, 185(2):604–613, feb 2014.

[21] Toni Giorgino. PLUMED-GUI: An environment for the interactive development of molecular dynamics analysis and biasing scripts. *Computer Physics Communications*, 185(3):1109–1114, mar 2014.

[22] Xevi Biarnés, Fabio Pietrucci, Fabrizio Marinelli, and Alessandro Laio. METAGUI. a VMD interface for analyzing metadynamics and molecular dynamics simulations. *Computer Physics Communications*, 183(1):203–211, jan 2012.

[23] Toni Giorgino, Alessandro Laio, and Alex Rodriguez. METAGUI 3: A graphical user interface for choosing the collective variables in molecular dynamics simulations. *Computer Physics Communications*, 217:204–209, aug 2017.

[24] *Collective Variables Module: Reference Manual for VMD. The Colvars Dashboard.*, 2021.

[25] Reza Salari, Thomas Joseph, Ruchi Lohia, Jérôme Hénin, and Grace Brannigan. A streamlined, general approach for computing ligand binding free energies and its application to gpcr-bound cholesterol. *Journal of Chemical Theory and Computation*, 14(12):6560–6573, 2018.

[26] J. Hénin. Fast and accurate multidimensional free energy integration. *J. Chem. Theory Comput.*, 2021. in press.

[27] Massimo Marchi and Pietro Ballone. Adiabatic bias molecular dynamics: A method to navigate the conformational space of complex molecular systems. *J. Chem. Phys.*, 110(8):3697–3702, 1999.

[28] Frédéric Cérou, Arnaud Guyader, and Mathias Rousset. Adaptive multilevel splitting: Historical perspective and recent results. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 29(4):043108, April 2019.

[29] Elise Duboué-Dijon and Jérôme Hénin. Building intuition for binding free energy calculations: Bound state definition, restraints, and symmetry. *The Journal of Chemical Physics*, 154(20):204101, 2021.

[30] Mina Ebrahimi and Jérôme Hénin. Symmetry-adapted restraints for binding free energy calculations. 2021. in preparation.

[31] Giacomo Fiorin, Fabrizio Marinelli, and José D. Faraldo-Gómez. Direct derivation of free energies of membrane deformation and other solvent density variations from enhanced sampling molecular dynamics. *J. Comp. Chem.*, 41(5):449–459, 2020.

[32] Leonardo G. Trabuco, Elizabeth Villa, Kakoli Mitra, Joachim Frank, and Klaus Schulten. Flexible fitting of atomic structures into electron microscopy maps using molecular dynamics. *Structure*, 16(5):673–683, may 2008.

[33] Abhishek Singharoy, Ivan Teo, Ryan McGreevy, John E Stone, Jianhua Zhao, and Klaus Schulten. Molecular dynamics-based refinement and validation for sub-5 å cryo-electron microscopy maps. *Elife*, 5:e16105, 2016.

[34] John W. Vant, Daipayan Sarkar, Giacomo Fiorin, Robert Skeel, Josh V. Vermaas, and Abhishek Singharoy. Data-guided multi-map variables for ensemble refinement of molecular movies. *bioRxiv*, 2020.

[35] Rudresh Acharya, Vincenzo Carnevale, Giacomo Fiorin, Benjamin G. Levine, Alexei L. Polishchuk, Victoria Balannik, Ilan Samish, Robert A. Lamb, Lawrence H. Pinto, William F. DeGrado, and Michael L. Klein. Structure and mechanism of proton transport through the transmembrane tetrameric m2 protein bundle of the influenza a virus. *Proceedings of the National Academy of Sciences*, 107(34):15075–15080, 2010.

[36] Eleonora Gianti, Vincenzo Carnevale, William F. DeGrado, Michael L. Klein, and Giacomo Fiorin. Hydrogen-bonded water molecules in the m2 channel of the influenza a virus guide the binding preferences of ammonium-based inhibitors. *The Journal of Physical Chemistry B*, 119(3):1173–1183, 2015. PMID: 25353315.

[37] *Preparing and using Multi-Map collective variables.*, 2020.