Elsevier Editorial System(tm) for Computer Physics Communications
Manuscript Draft

Manuscript Number:

Title: PLUMED-GUI: an environment for the interactive development of molecular dynamics analysis and biasing scripts

Article Type: Computer programs in physics paper

Keywords: Graphical User Interface; VMD; PLUMED; Molecular Dynamics; Collective Variables; Metadynamics

Corresponding Author: Dr. Toni Giorgino , PhD

Corresponding Author's Institution: National Research Council of Italy

First Author: Toni Giorgino , PhD

Order of Authors: Toni Giorgino , PhD

Abstract: PLUMED-GUI is an interactive environment to iteratively develop and test complex PLUMED scripts within the Visual Molecular Dynamics (VMD) environment. Computational biophysicists can take advantage of the best of VMD and PLUMED, leveraging a rich syntax to define collective variables (CVs), VMD's chemically-aware atom selection language, coding within a natural point-and-click interface. Pre-defined templates and syntax mnemonics provide support for inserting well-known reaction coordinates. Complex CVs, e.g. involving reference snapshots used for RMSD or native contacts calculations, can be built through dialogs that provide a synoptic view of the available options. Scripts can be either exported for use in simulation programs, or evaluated on the currently-displayed molecular trajectories. All of the script development takes place without leaving VMD, thus supporting an incremental try-see-modify development cycle for molecular metrics.

Suggested Reviewers: John E Stone Dr.
Senior Research Programmer, Theoretical and Computational Biophysics Group, University of Illinois at Urbana-Champaign
johns@ks.uiuc.edu
Dr. Stone is one of the architects and developers of VMD, the widely-used molecular modeling and analysis environment of which this package is an extension.

Jan Saam Dr.
Researcher, Theoretical and Computational Biophysics Group, University of Illinois at Urbana-Champaign
saam@ks.uiuc.edu
Dr. Saam is one of the authors of VMD and numerous plugins in that environment, including the “multiplot plugin” used by Plumed-GUI to plot visualize the collective variable values and export the results.

Olaf Lenz Dr. rer. nat.
Researcher, Institut für Computerphysik, Universität Stuttgart
olenz@icp.uni-stuttgart.de

Dr. Lenz has made contributions to various scientific software packages, for example the simulation packages ESPResSo++ and ESPResSo, and the molecular viewer VMD.

Giovanni Bussi Dr.
Researcher, Molecular and Statistical and Biophysics, Scuola Internazionale Superiore di Studi Avanzati (SISSA)
bussi@sissa.it
Corresponding author of the PLUMED 2 companion paper, the computational engine on which this GUI is based.

ISIB CNR
Consiglio Nazionale delle Ricerche

Istituto di Ingegneria Biomedica
corso Stati Uniti, 4
35127 Padova
tel: +39.049.8295702
fax: +39.049.8295763
email: mbox@isib.cnr.it
codice fiscale: 80054330586
partita IVA: 02118311006

Sezione di Milano
Politecnico di Milano
c/o Dip. di Bioingegneria
piazza Leonardo da Vinci, 32
20133 Milano

Sezione di Roma
Area di Ricerca Roma 1 – Montelibretti
via Salaria Km 29,300
00016 Monterotondo S. (RM)

Padua, 31st of July, 2013

Dear Prof. Truhlar,

please find enclosed a manuscript, titled *Plumed-GUI: an environment for the interactive development of molecular dynamics analysis and biasing scripts.* The manuscript describes the rationale and use of Plumed-GUI, a graphical interface integrating two well known codes, namely Plumed (computation core) and Visual Molecular Dynamics (visualization).

The GUI constitutes a front-end to the current version of the PLUMED engine, an actively-developed system for computing a variety of reaction coordinates in biomolecular simulations, itself integrated with most current molecular dynamics engines. The scope of the Plumed-GUI is to allow scientists to efficiently iterate, evaluate and revise scripts composed of (often complex) collective variable definitions, delegating their computation to the PLUMED engine. Such scripts can either be run to analyze the results of biomolecular simulations, or be exported to become the basis of force-biasing protocols.

Both PLUMED and Plumed-GUI are mature and in active use: Plumed-GUI is packaged with VMD since about a year, but had not been formalized in a "suggested publication" yet; PLUMED (version 2.0) has been recently released, and is described in a manuscript submitted to your Journal at the same time as this[1]. Even if distinct (and with distinct authorship) the two packages are kept in sync and are complementary in scope; analogously, we feel that the two papers complement each other well. Therefore, in agreement with Plumed's authors, here we would like propose that the two papers are considered for publication in the same issue of the Computer Physics Communications. I believe that they will be valuable and interesting contributions for a wide readership of your Journal.

Best regards,

Toni Giorgino
Institute for Biomedical Engineering (ISIB)
National Research Council of Italy (CNR)
Corso Stati Uniti 4
I-35127 Padova
Italy

Email: toni.giorgino@isib.cnr.it
Phone: +39 049 8295761
Fax: +39 049 8295763

1 G. A. Tribello, M. Bonomi, D. Branduardi, C. Camilloni, G. Bussi, PLUMED 2: New feathers for an old bird.

Suggested reviewers:

- First, the authors of the PLUMED engine are obviously entitled to comment on the suitability of the Plumed-GUI for the general public, and the clarity of the description set forth in this manuscript. The Plumed developers received an early draft of this manuscript.

- Dr. John E Stone
  Senior Research Programmer
  Theoretical and Computational Biophysics Group, University of Illinois at Urbana-Champaign
  johns@ks.uiuc.edu
  Dr. Stone is one of the architects and developers of VMD, the widely-used molecular modeling and analysis environment of which this package is an extension.

- Dr. rer. nat. Olaf Lenz
  Institut für Computerphysik, Universität Stuttgart, Allmandring 3, D-70569 Stuttgart
  olenz@icp.uni-stuttgart.de
  Dr. Lenz has made contributions to various scientific software packages, for example the simulation packages ESPResSo++ and ESPResSo, and the molecular viewer VMD.

- Dr. Jan Saam
  Theoretical and Computational Biophysics Group, Beckman Institute, University of Illinois at Urbana-Champaign
  saam@ks.uiuc.edu
  Dr. Saam is one of the authors of VMD and numerous components of that environment, including the “multiplot plugin” used by Plumed-GUI to plot visualize the collective variable values and export the results.

PLUMED-GUI: an environment for the interactive development of molecular dynamics analysis and biasing scripts

Toni Giorgino*

Institute of Biomedical Engineering (ISIB), National Research Council of Italy (CNR), Corso Stati Uniti 4, I-35127 Padua, Italy

Abstract

PLUMED-GUI is an interactive environment to iteratively develop and test complex PLUMED scripts within the Visual Molecular Dynamics (VMD) environment. Computational biophysicists can take advantage of the best of VMD and PLUMED, leveraging a rich syntax to define collective variables (CVs), VMD's chemically-aware atom selection language, coding within a natural point-and-click interface. Pre-defined templates and syntax mnemonics provide support for inserting well-known reaction coordinates. Complex CVs, e.g. involving reference snapshots used for RMSD or native contacts calculations, can be built through dialogs that provide a synoptic view of the available options. Scripts can be either exported for use in simulation programs, or evaluated on the currently-displayed molecular trajectories. All of the script development takes place without leaving VMD, thus supporting an incremental try-see-modify development cycle for molecular metrics.

Keywords: Graphical User Interface, VMD, PLUMED, Molecular Dynamics, Collective Variables, Metadynamics

*To whom correspondence should be addressed
Email address: toni.giorgino@isib.cnr.it (Toni Giorgino)

Program summary

Manuscript Title: PLUMED-GUI: an environment for the interactive development of molecular dynamics analysis and biasing scripts

Authors: Toni Giorgino

Program Title: PLUMED-GUI (Collective variable analysis plugin)

Journal Reference:

Catalogue identifier:

Licensing provisions: 3-clause BSD Open Source.

Programming language: TCL/TK.

Operating system: Linux/Unix, OSX, Windows.

RAM: Sufficient to run PLUMED [1] and VMD [2].

Number of processors used: 1

Keywords: Graphical User Interface VMD PLUMED Molecular Dynamics Collective Variables Metadynamics

Classification: 3 Biology and Molecular Biology, 23 Statistical Physics and Thermodynamics.

Subprograms used: PLUMED (version 1.3 or higher).

Nature of problem: Compute and visualize values of collective variables on molecular dynamics trajectories from within VMD, and interactively develop biasing scripts for the estimation of free-energy surfaces in PLUMED.

Solution method: A graphical user interface is integrated in VMD and allows to interactively develop and run analysis scripts. Menus and dialogs provide mnemonics and documentation on the syntax to define complex CVs.

Restrictions: Tested on systems up to 100,000 atoms.

Unusual features: VMD-PLUMED is not a standalone program but a plugin that provides access to PLUMED's analysis features from within VMD.

Additional comments: Distributed with VMD since version 1.9.0. Manual update may be required to access the latest features.

Running time: Computations of the values of collective variables, performed by the

underlying PLUMED code, depends on the size of the system and the length of the trajectory; it is generally negligible with respect to simulation time.

1. Introduction

Molecular dynamics (MD) is a computational technique which models the interactions between a set of atoms with realistic empirical potentials. Recent increases in computer power allow to routinely sample biomolecular systems with all-atom resolution for biologically-relevant timescales, thus providing *in silico* approximated views on processes that are too fast, or too small to be measured *in vitro*. Recent examples include protein folding [3], channel permeation and gating [4], drug binding [5, 6], protein-protein interactions [7, 8], and so on, not to mention applications in materials science and coarse-grained macromolecular assemblies.

An atomistic molecular model involves thousands to millions of degrees of freedom, which are hardly interpretable directly. Biophysically or biochemically relevant information, such as free energies, kinetic rates, transition probabilities, and so on, is usually extracted aggregating relevant degrees of freedom into reaction coordinates or *collective variables* (CVs), defined as mathematical functions of (some of) the coordinates of the system. CVs thus simplify the interpretation of complex events, and are normally used as independent coordinates in formalisms such as the potential of mean force.

Choosing a set of CVs to adequately describe a given system is, however, not trivial. In general, it is important to identify those reaction coordinates which change "slowly" over the timescales of the phenomena of interest. CVs thus identified can then be monitored to detect rare events [8], be biased to determine free energy landscapes [9], used to partition the phase space to

reconstruct kinetic rates [10, 11], and so on. Although chemical intuition is a guide in the selection of CVs, some amount of tuning is generally required in parametrizing the specific details of the functions.

Several software packages offer the possibility to compute CVs; however, existing software is usually restrictive on the complexity of the functions that can be defined, limited to the analysis phase, or requires users to explicitly code the CV computations in ad-hoc scripts, which therefore tend to contain "boilerplate" code that obfuscates the metric. To the contrary, it would be desirable to have a concise and human-readable definition of both the functional form (e.g., "distance", "contacts", "interfacial waters", ...) and the atoms involved (say, "protein", "charged residues", "molecules close to residue X", ...).

A step forward in this direction is PLUMED, a flexible CV engine recently upgraded to version 2.0 [1]. PLUMED provides an extensive set of pre-defined *actions*, i.e. self-explanatory keywords that concisely define a CV on the basis of the geometry of a system. Auxiliary actions also exist to define center of masses, ghost atoms, units, etc. [12, 13] PLUMED scripts, in general, contain actions to define several CV, plus, if desired, statements that express the biasing protocol to be employed during simulation. The values of CVs can also be computed on existing trajectories (trajectory analysis) through its *driver* feature.

This paper introduces PLUMED-GUI, a plugin integrated with the widely-used Visual Molecular Dynamics (VMD) molecular analysis and visualization software [2] to streamline the development and test of analysis scripts. Together, PLUMED and PLUMED-GUI offer a concise and homogeneous way to express CVs and evaluate them; VMD provides intuitive facilities to load and visualize the trajectories under analysis, an easy to use graphical

environment, and a powerful, topology-aware atom selection language for selecting molecular components.

2. Plugin usage

PLUMED-GUI is started selecting the *Analysis/Collective variable analysis (Plumed)* entry in VMD's *Extensions* menu. The main text area hosts the PLUMED script, entered following the syntax of the PLUMED version currently in use (Figure 1(a)). The interface behaves as a text editor; *File* and *Edit* menus provide customary editing commands, including open and save, copy/paste and undo/redo operations. Initially, the text area displays a brief syntax reminder, which can be dismissed.

It is worthwhile noting that the GUI does not restrict the input syntax. The script is passed as-is to the underlying PLUMED engine, with the sole exception of symbolic atom selections in square brackets, which are resolved as will be shown in Section 3.1. Script coding and debug is entirely under the control of the user, and therefore any valid or invalid expression can be entered. (Consequently, the GUI needs no updates to accommodate user-customized PLUMED variants and future syntax.)

2.1. Analysis and visualization

Pressing the *Plot* button at the bottom of the window evaluates the displayed script on the currently selected trajectory (known within VMD as the all-important *top* molecule). Assuming that PLUMED executable is properly installed, the GUI will execute PLUMED's *driver* function, which will evaluate the values of the CVs defined in the script at each of the top trajectory frames.

Once the evaluation is successful, the time series of the collective variables are displayed graphically in a plot. The purpose of the plot is to quickly inspect the values yielded by the current CV definitions, and provide a way to iteratively refine them. The plot layout shows time on the abscissa and the CV values in different line styles; data points can be optionally read out hovering the mouse pointer. More complex visualizations can be obtained exporting data to external plotting programs; data can be exported either as a matrix (time running as rows, and CVs as columns), or as consecutive time-value vectors separated by empty lines.

Should the evaluation of the script generate an error, it will be displayed in the VMD textual console. In most instances PLUMED identifies the specific problem and corresponding script line; when this happens, the error line will be highlighted as such in the text area.

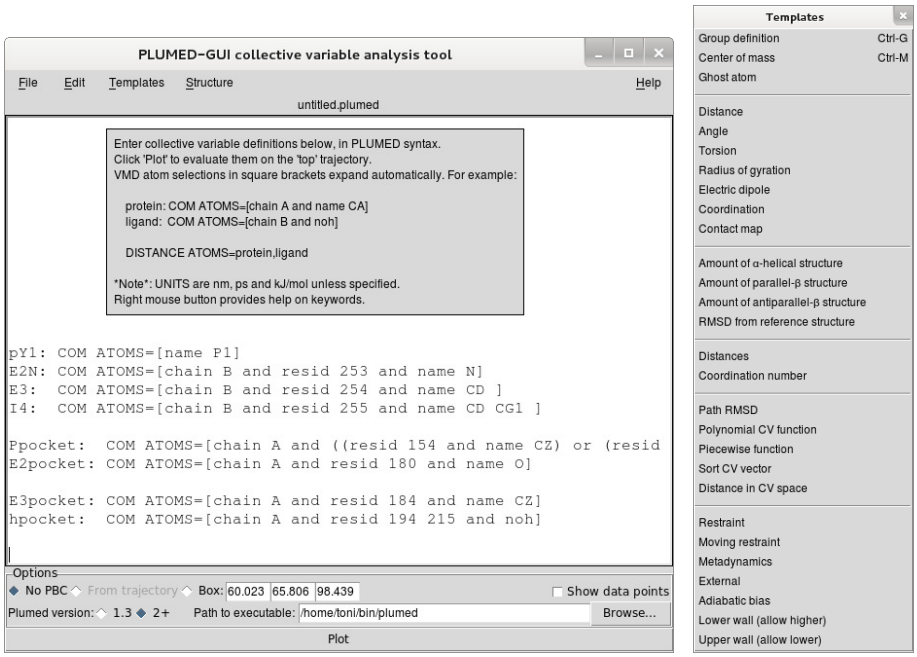
2.2. Consistency of units

It may be worth noting that the units of computed CVs depend on PLUMED's conventions. Since version 2.0, PLUMED defaults to the nm, kJ/mol, ps combination. Given that VMD users may be accustomed to the Å, kcal/mol, fs unit set, a reminder is shown about the fact that the `UNITS` keyword can be used at the top of the script to switch to customary units.

3. Assisted script development

3.1. Symbolic atom selections

VMD users are usually familiar with the program's powerful language for atom selections; strings such as `same residue as (protein or water within 4 of name CA)` are useful expressions that are interpreted at run time, and are equivalent to a list of atoms. The sophisticated syntax can



(a) Main window (b) Templates menu

Figure 1: (a) PLUMED-GUI's main window. The analysis script is entered in the text area, like a text editor. The *Plot* button evaluates the collective variables defined in the script on the molecular trajectory currently selected in VMD ("top molecule"); if successful, a graph appears, showing the values of the CVs at each frame. The inner gray box, only shown at startup, is a brief reminder about the use of the interface. (b) The *Templates* menu contains shortcuts that insert frequently-used definitions and collective variables.

query atoms on the basis of numerical (coordinates, beta values, residue IDs), chemical (e.g. polar, atom names) and/or other properties, as documented elsewhere [2].

PLUMED-GUI allows the use of VMD's atom selections in PLUMED scripts through square brackets. As shown in Figure 1(a), bracketed textual expressions are evaluated with respect to the current frame of the top molecule, and transparently replaced with the resulting list of atoms. In this way, PLUMED users can avoid the use of numeric atom IDs altogether in favor of human-readable expressions such as `[protein and name CA]`.

This is especially advantageous when switching between multiple systems; it is the case, for example, when several all-atom systems are prepared containing same protein and a series of compounds. Whereas atom indices depend on the specific system and the details of how it was prepared, expressions such as `[not protein and not water]` (matching non-peptide ligands) do not, and will be valid regardless of the specific system being analyzed.

Symbolic atom expressions are interpreted at the moment the analysis is started by pressing the *Plot* button. They can also be permanently replaced with atom numbers to be used independently of PLUMED-GUI, via the *Export* function presented in section 5.

3.2. Templates

The *Templates* menu provides shortcuts that insert a number of frequently-used definitions; selecting one of the menu entries types the corresponding keyword in the text area at the cursor's position (Figure 1(b)). Templates, in other words, offer human-readable shortcuts to enter the frequently used strings that define atom groups and CVs. After insertion, templates can be

edited freely in the text area. Templates have to be filled in manually; for example, in the case of the "Coordination" template, one has to specify one or two groups between which the coordination number is to be computed, and the parameters of the switching function.

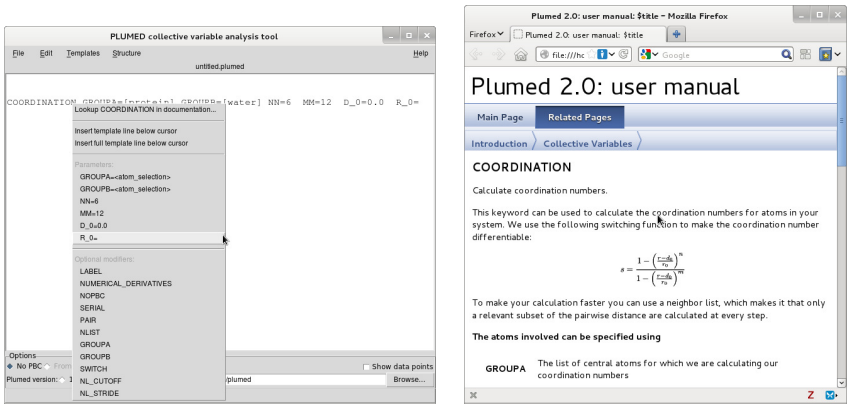
The list of templates provided in the menu is not meant to be exhaustive, but rather to provide a synopsis of to the most frequently-used CVs, inserted with the default options. Generic actions and modifiers can be typed manually, while optional keywords can be looked up through an on-line contextual help, described in the next section.

3.3. On-line help

PLUMED's actions have a wealth of options to alter the behavior of CVs. For instance, the `COORDINATION` action foresees modifiers to define the shape and functional form of the switching function; to ignore periodic boundary conditions; to compute derivatives numerically; and several others. The richness of the syntax may make it unwieldy to recall the syntax of lesser-used options.

To this end, PLUMED-GUI provides a comprehensive context-dependent help facility through a pop-up menu, which is be invoked pressing the right mouse button on any action keyword. The topmost menu item, *Lookup in documentation*, opens up a web browser displaying the full manual page of that action. Subsequent entries in the pop-up menu shows the list of optional and mandatory modifiers accepted by that action 2.

As for the rest of PLUMED 2.0 documentation, PLUMED-GUI's contextual help is generated automatically from PLUMED's source code. This implies that, as long as new features are implemented and documented according to the established coding conventions, any newly-developed func-



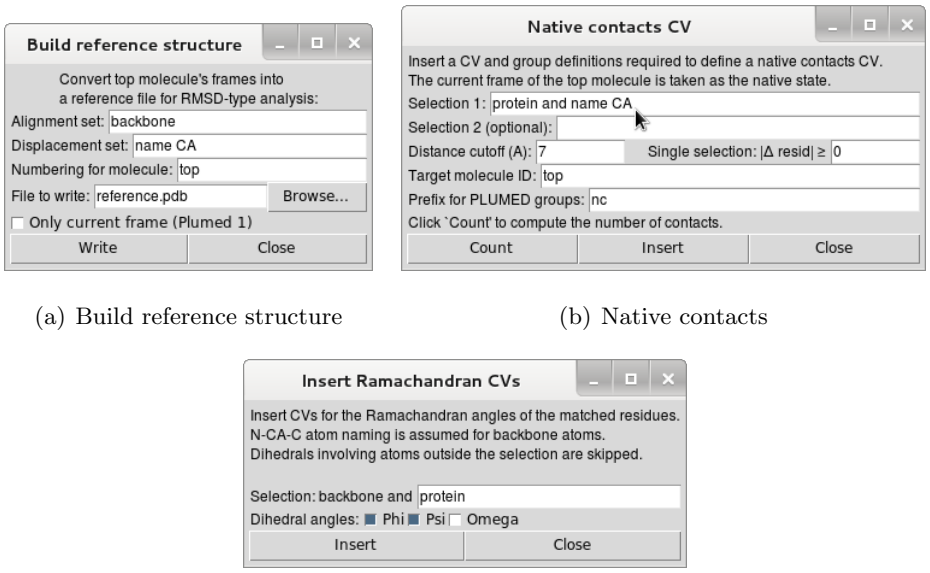
(a) Contextual help (b) Sample manual page

Figure 2: (a) A contextual popup menu lists mandatory and optional keywords supported by the action under the pointer (in this case, `COORDINATION`, which computes the coordination number of one or two groups of atoms). (b) The *Lookup* function recalls an action's manual in the web browser.

tions become properly integrated in the interface, without requiring modifications to the GUI code.

4. Structure-based operations

Functions in the *Structure* menu provide assistance in the definition of more complex CVs that depend upon the topology and coordinates of the currently loaded system. Each of the menu entries opens up a dialog with a number of tunable options. Structure-based CVs generally involve long lists of statements and/or auxiliary files; these automated procedures are meant to relieve users from the error-prone process of building files and lists by hand.



(a) Build reference structure

(b) Native contacts

(c) Backbone torsion angles

Figure 3: Dialogs accessible from the *Structure* menu support the creation of CVs based on the active topology. (a) *Build reference structure* converts the currently displayed frame into a reference file for RMSD calculations. Atom sets to be used for alignment and displacement are specified as VMD atom selections; numbering can also be mapped between molecules if the reference frame and the trajectory on which the CV will be computed belong to systems with different topologies. (b) Analogously, *Native contacts* enumerates the atom pairs (closer than the chosen threshold distance) in the currently-displayed ("native") frame. The CV will measure how many of those atom pairs will present in each trajectory frame. Non-informative contacts between neighboring residues can be filtered out putting a lower bound to the $|\Delta\text{resid}|$ parameter. (c) The *Insert backbone angles* dialog inserts CVs corresponding to ϕ, ψ and/or ω dihedrals contained in the selection.

4.1. Generating reference structures for alignments

The root mean square deviation (RMSD) metric is frequently used to detect structural similarities and conformational transitions. RMSD values are computed averaging the squared displacement of a chosen set of atoms (displacement set) with respect to a *reference* structure, after applying the roto-translation that optimally aligns another, possibly coincident, set of atoms (alignment set). PLUMED also implements three generalization of the metric, namely the S, Z and *property map* path variables, to express the "progression" and "distance" of the current state of the system along a path defined by an arbitrary number of exemplary reference structures used as landmarks [14, 15].

The *Build reference structure* dialog provides a convenient way to generate such reference structures (Figure 3(a)). Pressing the *Write* button "freezes" the coordinates of the *currently selected* frame into a "reference file". Reference files are PDB-like tables used by PLUMED to define the set of atoms to be used for alignment, for computing the displacement, and the reference coordinates; each line represents one of the atoms involved in the calculation, with columns recording serial numbers, coordinates, and inclusion in one or the other set [12].

The dialog allows the use of atom selections to indicate the subset of the atoms to be involved, respectively, in the computation of the optimal alignment, and the measure of the RMSD. A check-box provides a choice on whether to export all of the frames of the current trajectory (convenient when specifying a complex path), or just the current frame (for basic RMSD calculations, or to facilitate the manual construction of paths).

By default, the reference file generated is suitable for computing S, Z and property map values on systems with the same topology as the one

from which the reference was extracted. However, it is sometimes necessary to perform alignments between different topologies; for example, the native structure may be a PDB file, while the system under analysis is the all-atom structure used in simulation. Alignments between molecules with different topologies are possible by setting the *target molecule ID*. This feature adjusts the atom numbering of the top molecule to be compatible with the specified *target* molecule; in other words, trajectory frames of the target molecule will be aligned with the structure of the top molecule, even though the topologies of the two are different. The renumbering feature requires that the atom selections match the same number of atoms in the two systems.

4.2. Number of native contacts

The number of native contacts is another metric to determine structural similarity, frequently used as an indicator of folding or binding. The metric puts the accent on the presence of those contacts that characterize the desired (native) structure. First, the pairs of atoms in contact in a given native structure are enumerated. Then, this list is evaluated for each of the trajectory frames under analysis: the CV counts how many of the pairs that were in contact in the reference frame are also close in the frame being analyzed.

The *Native contacts* dialog (Figure 3(b)) can be used to generate such lists flexibly and with ease. Like when building reference structures, the current frame of the top molecule is used as the native state. It is possible to specify either one or two atom selections; in the first case, the contacting pairs involving atoms in the selection are enumerated; otherwise, if two selections are given, intermolecular contacts – bridging the two selections – will be counted. The "distance cutoff" box adjusts the distance (in Å) at

which an atom pair is assumed to be in contact.

A marked rise in the number of native contacts is often used as a proxy for the detection of folding events. However, residues adjacent in the primary sequence will almost always be in contact, thus contributing little or no information to the folding signal. These "trivial" contacts can be filtered out setting a minimum bound to the $|\Delta\text{resid}|$ to a positive integer d. If set, contacts between atoms closer than d residues apart in the primary sequence will be disregarded. Analogously to the *Build reference structure* function, the user can match a trajectory with a native frame with a different topology specifying the appropriate target molecule ID.

The number of native contacts is implemented in PLUMED through the `COORDINATION PAIRS` action and the enumeration of the contacting pairs in the native frame. It is worthwhile noting that, like all other CVs provided by PLUMED, this metric is a continuous approximation of the integer pair count, made smooth with respect to all of the system's coordinates through an exponential switching function [12].

4.3. Backbone torsion angles

The *Insert backbone angles* dialog (Figure 3(c)) allows the computation of backbone ϕ, ψ and/or ω torsion angles between neighboring residues, defined according to the standard IUPAC rules for biochemical nomenclature [16]. The user is asked to specify an atom selection; when the *Insert* button is pressed, a CV will be inserted for each ϕ, ψ and/or ω backbone dihedral contained in the selection. Each angle is defined through the appropriate `TORSION` keyword and, for the sake of readability, includes a comment pointing back to the name of the involved residue.

5. Export for use in simulation

PLUMED has extensive facilities to biases molecular dynamics simulations with forces that enhance the sampling of the phase-space in a way that allows the reconstruction of free-energy surfaces. Example of biasing protocols include harmonically constraining CVs at a given combination of values (used e.g. for the umbrella sampling protocol [17]), pulling them towards increasing or decreasing values (steered MD [18, 19]), metadynamics [9], and so on. Biased MD simulations are carried out with codes patched to embed the PLUMED engine. Force biases are specified in the script, which defines the biasing protocol as well as the CVs to be biased. Atoms have to be specified through their serial numbers, which makes the iteration of complex scripts through different systems an error-prone exercise.

The *Export* function, accessible from the File menu, removes all the symbolic atom selections in the current script and replaces them with the corresponding numerical lists. The exported script is thus devoid of VMD-specific constructs, and can then be employed for simulations. The exported file contains comments to remind how the numeric atom lists were obtained although, for the sake of reproducibility, it is generally advisable to keep the original script with unsubstituted, symbolic atom selections.

6. Installation and compatibility

The GUI supports the same wide range of platforms as VMD, encompassing all major variants of Linux/Unix, OSX, and Windows. Trajectory analysis is performed invoking the platform-specific *driver* executable behind the scenes. PLUMED distributions provide instructions on how to

build the executable on Unix-like systems; a precompiled version for Windows is provided for convenience.

Recent VMD distributions contain a preinstalled version of PLUMED-GUI. This paper describes version 2.0 of the GUI, which supports PLUMED 2.0 and earlier. Users may manually update their GUI to the latest version. To update the GUI, it is sufficient to identify VMD's installation path, and replace the files contained in the subdirectory `plugins/noarch/tcl/plumed` with those contained in the latest distribution of PLUMED-GUI. As customary, the *About* menu item displays the currently installed version of the GUI.

For clarity, this paper focuses on the features available with PLUMED 2.0. The current version of the plugin, PLUMED-GUI 2.0, supports both PLUMED 1.3 and PLUMED 2.0, with minor functional differences. Language syntax and *driver* invocation method differ between the two PLUMED versions. The GUI detects which version is installed and adapts templates and syntax accordingly. If both PLUMED versions are available, the user can switch manually between the two.

7. Conclusions

Developing an appropriate combination of reaction coordinates is a central task in the analysis of biomolecular systems. PLUMED-GUI simplifies the iterative development, refinement and test of collective variables to be used with the PLUMED engine. The GUI bridges the usability of VMD's graphical interface and PLUMED's rich CV definition language.

Integrating the two environments incurs in a few limitations; right now, only orthorhombic simulation boxes with constant edges are supported,

therefore precluding the analysis of constant-pressure simulations (this limitation may be removed as soon as *driver*'s support to trajectory formats is expanded). Another drawback is due to the fact that atom selections are evaluated only once, before the computation is started; thus, it is not possible to employ time-varying atom lists (nor PLUMED engine would support them): analysis protocols involving time-varying atom sets are outside of the scope of the programs. It is worthwhile noting, however, that PLUMED 2 provides switching functions (such as `DISTANCES LESS_THAN`) that are continuous approximations to discrete quantities such as the number of atoms satisfying a given property.

One of the objectives of the GUI is to lower the barrier for the adoption of meaningful metrics in the analysis tasks of simulation data. In the future, the interface may be expanded integrating more "function building" features and providing interfaces with external programs, such as METAGUI [10] and reweighting schemes [20].

8. Acknowledgments

I would like to thank the authors of PLUMED and VMD for creating, distributing and supporting the corresponding software packages. An acknowledgment goes to Prof. G. De Fabritiis and colleagues at the Computational Biophysics Laboratory at the Universitat Pompeu Fabra (Barcelona), where this work was started. Former support from the "Beatriu de Pinós" scheme of the Agència de Gestió d'Ajuts Universitaris i de Recerca (Generalitat de Catalunya) is gratefully acknowledged.

References

[1] G. A. Tribello, M. Bonomi, D. Branduardi, C. Camilloni, G. Bussi, PLUMED 2: New feathers for an old bird, Computer Physics Communications – This issue.

[2] W. Humphrey, A. Dalke, K. Schulten, VMD: visual molecular dynamics, J Mol Graph 14 (1) (1996) 33–38.

[3] K. Lindorff-Larsen, S. Piana, R. O. Dror, D. E. Shaw, How fast-folding proteins fold, Science 334 (6055) (2011) 517–520. doi:10.1126/science.1208351.

[4] M. O. Jensen, V. Jogini, D. W. Borhani, A. E. Leffler, R. O. Dror, D. E. Shaw, Mechanism of voltage gating in potassium channels, Science (New York, N.Y.) 336 (6078) (2012) 229–233, PMID: 22499946. doi:10.1126/science.1216533.

[5] Y. Shan, E. T. Kim, M. P. Eastwood, R. O. Dror, M. A. Seeliger, D. E. Shaw, How does a drug molecule find its target binding site?, Journal of the American Chemical Society 133 (24) (2011) 9181–9183. doi:10.1021/ja202726y.

[6] I. Buch, T. Giorgino, G. De Fabritiis, Complete reconstruction of an enzyme-inhibitor binding process by molecular dynamics simulations, Proceedings of the National Academy of Sciences 108 (25) (2011) 10184–10189. doi:10.1073/pnas.1103547108.

[7] M. Ahmad, W. Gu, V. Helms, Mechanism of fast peptide recognition by sh3 domains, Angewandte Chemie (International Ed. in En-

glish) 47 (40) (2008) 7626–7630, PMID: 18752238. doi:10.1002/anie.200801856.

[8] T. Giorgino, I. Buch, G. De Fabritiis, Visualizing the induced binding of sh2-phosphopeptide, J. Chem. Theory Comput. 8 (4) (2012) 1171–1175. doi:10.1021/ct300003f.

[9] A. Laio, M. Parrinello, Escaping free-energy minima, Proceedings of the National Academy of Sciences of the United States of America 99 (20) (2002) 12562–12566, PMID: 12271136. doi:10.1073/pnas.202427399.

[10] X. Biarnés, F. Pietrucci, F. Marinelli, A. Laio, Metagui. a vmd interface for analyzing metadynamics and molecular dynamics simulations, Computer Physics Communications 183 (1) (2012) 203–211. doi:10.1016/j.cpc.2011.08.020.

[11] F. Noé, S. Fischer, Transition networks for modeling the kinetics of conformational change in macromolecules, Current Opinion in Structural Biology 18 (2) (2008) 154–162, PMID: 18378442. doi:10.1016/j.sbi.2008.01.008.

[12] Plumed user's guide. a portable plugin for free-energy calculations with molecular dynamics.

[13] M. Bonomi, D. Branduardi, G. Bussi, C. Camilloni, D. Provasi, P. Raiteri, D. Donadio, F. Marinelli, F. Pietrucci, R. A. Broglia, M. Parrinello, PLUMED: a portable plugin for free-energy calculations with molecular dynamics, Computer Physics Communications 180 (10) (2009) 1961–1972. doi:10.1016/j.cpc.2009.05.011.

[14] D. Branduardi, F. L. Gervasio, M. Parrinello, From A to B in free energy space, The Journal of Chemical Physics 126 (5) (2007) 054103. doi:10.1063/1.2432340.

[15] V. Spiwok, B. Králová, Metadynamics in the conformational space nonlinearly dimensionally reduced by isomap, The Journal of Chemical Physics 135 (22) (2011) 224504–224504–6. doi:10.1063/1.3660208.

[16] IUPAC-IUB Commission on Biochemical Nomenclature, Abbreviations and symbols for the description of the conformation of polypeptide chains. tentative rules (1969), Biochem J 121 (4) (1971) 577–585.

[17] G. M. Torrie, J. P. Valleau, Nonphysical sampling distributions in monte carlo free-energy estimation: Umbrella sampling, Journal of Computational Physics 23 (2) (1977) 187 –199. doi:10.1016/0021-9991(77)90121-8.

[18] B. Isralewitz, M. Gao, K. Schulten, Steered molecular dynamics and mechanical functions of proteins, Current Opinion in Structural Biology 11 (2) (2001) 224–230. doi:10.1016/S0959-440X(00)00194-9.

[19] T. Giorgino, G. De Fabritiis, A high-throughput steered molecular dynamics study on the free energy profile of ion permeation through Gramicidin A, J. Chem. Theory Comput. 7 (6) (2011) 1943–1950. doi:10.1021/ct100707s.

[20] M. Bonomi, A. Barducci, M. Parrinello, Reconstructing the equilibrium boltzmann distribution from well-tempered metadynamics, Journal of Computational Chemistry 30 (11) (2009) 16151621. doi:10.1002/jcc.21305.