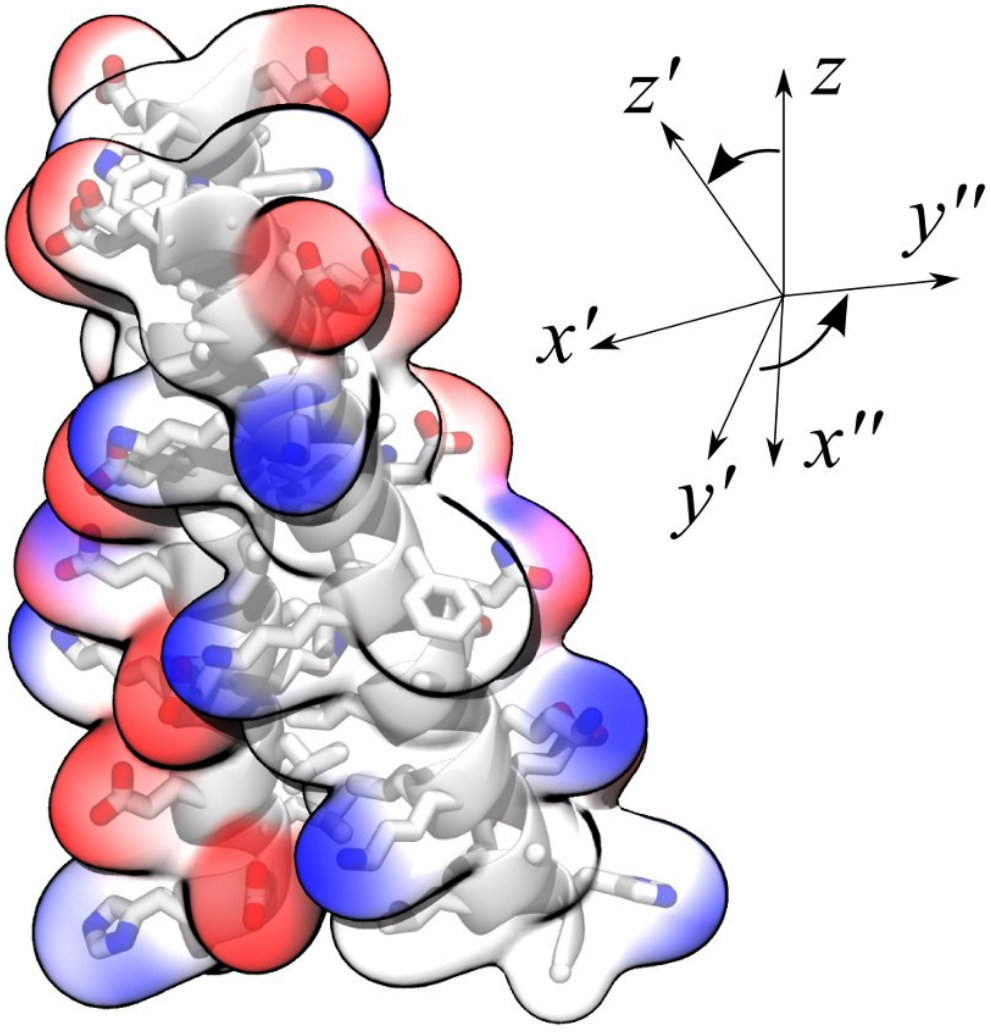
COLLECTIVE VARIABLES MODULE
Reference manual for LAMMPS

Code version: 2017-09-18



Giacomo Fiorin, Jérôme Hénin, Axel Kohlmeyer

Contents

1 Introduction

In molecular dynamics simulations, it is often useful to reduce the large number of degrees of freedom of a physical system into few parameters whose statistical distributions can be analyzed individually, or used to define biasing potentials to alter the dynamics of the system in a controlled manner. These have been called 'order parameters', 'collective variables', '(surrogate) reaction coordinates', and many other terms.

Here we use primarily the term 'collective variable' (shortened to *colvar*), which indicates any differentiable function of atomic Cartesian coordinates, x_i, with i between 1 and N, the total number of atoms:

$$\xi(t) = \xi\left(x_i(t), x_j(t), x_k(t), \ldots\right) \ , \ \ 1 \leq i, j, k \ldots \leq N \tag{1}$$

This manual documents the collective variables module (**Colvars**), a portable software that interfaces multiple MD simulation programs, with a focus on flexibility, robustness and high performance. The module is designed to perform multiple tasks concurrently during or after a simulation, the most common of which are:

- apply restraints or biasing potentials to multiple colvars, tailored on the system by choosing from a wide set of basis functions, without limitations on their number or on the number of atoms involved;
- calculate potentials of mean force (PMFs) along any set of colvars, using different enhanced sampling methods, such as Adaptive Biasing Force (ABF), metadynamics, steered MD and umbrella sampling; variants of these methods that make use of an ensemble of replicas are supported as well;
- calculate statistical properties of the colvars, such as running averages and standard deviations, correlation functions of pairs of colvars, and multidimensional histograms: this can be done either at run-time without the need to save very large trajectory files, or after a simulation has been completed using VMD and the `cv` command.

Detailed explanations of the design of the Colvars module are provided in reference [1]. Please cite this reference whenever publishing work that makes use of this module.

2 A crash course

Suppose that we want to run a steered MD experiment where a small molecule is pulled away from a binding site. In Colvars terms, this is done by applying a moving restraint to the distance between the two objects. The configuration will contain two blocks, one defining the distance variable (see section4, section6), and the other the moving harmonic restraint (subsection7.4).

```
colvar {
  name dist
  distance {
    group1 { atomNumbersRange 42-55 }
    group2 {
      psfSegID PR
      atomNameResidueRange CA 15-30 }
    }
  }
}

harmonic {
```

```
  colvars dist
  forceConstant 20.0
  centers 4.          # initial distance
  targetCenters 15.   # final distance
  targetNumSteps 500000
}
```

Reading this input in plain English: the variable here named *dist* consists in a distance function between the centers of two groups: the ligand (atoms 42 to 55) and the alpha carbon atoms (CA) of residues 15 to 30 in the protein (segment name PR). The atom selection syntax is detailed in section5.

To the "*dist*" variable, we apply a `harmonic` potential of force constant 20 energy units/Å^2, initially centered around a value of 4 Å, which will increase to 15 Å over 500,000 simulation steps.

3 General parameters and input/output files

Here, we document the syntax of the commands and parameters used to set up and use the Colvars module in LAMMPS. One of these parameters is the configuration file or the configuration text for the module itself, whose syntax is described in subsection3.2 and in the following sections.

3.1 LAMMPS keywords

To enable a Colvars-based calculation, the following line must be added to the LAMMPS configuration file:

`fix` *ID* `all colvars` *configfile keyword value pairs ...*

where *ID* is a string that uniquely identifies this fix command inside a LAMMPS script, *configfile* is the name of the configuration file for the Colvars module, followed by one or more of the following optional keywords with their corresponding arguments:

- `input` ⟨Name or prefix of the input state file⟩
  Context: Keyword of the `fix colvars` command
  Acceptable values: string
  Description: If a value is provided, it is interpreted as either the name of the input state file, or as the prefix of the file named *input*`.colvars.state`. This allows to continue a previous collective variables-based calculation when a regular binary LAMMPS restart file is not available (see subsection3.3).

- `output` ⟨Prefix of the output state file⟩
  Context: Keyword of the `fix colvars` command
  Acceptable values: string
  Default value: "out"
  Description: If a value is provided, it is interpreted as the prefix to all output files that will be written by the Colvars module (see subsection3.4).

- `unwrap` ⟨Whether to unwrap coordinates passed to the Colvars module⟩
  Context: keyword of the `fix colvars` command
  Acceptable values: "yes" or "no"
  Default value: "yes"

Description: This keyword controls whether wrapped or unwrapped coordinates are passed to the Colvars module for calculation of the collective variables and of the resulting forces. The default is to use the image flags to reconstruct the absolute atom positions: under this convention, centers of mass and centers of geometry are calculated as a weighted vector sum (see subsection5.3). Setting this to *no* will use the current local coordinates that are wrapped back into the simulation cell at each re-neighboring instead.

- `seed` ⟨Seed for the random number generator⟩
  Context: Keyword of the `fix colvars` command
  Acceptable values: positive integer
  Default value: 1966
  Description: If defined, the value of this keyword is provided as seed to the random number generator. This is only meaningful when the `extendedLangevinDamping` keyword is used (see subsection4.3).

- `tstat` ⟨Thermostating fix⟩
  Context: Keyword of the `fix colvars` command
  Acceptable values: string
  Default value: NULL
  Description: This keyword provides the *ID* of an applicable thermostating fix command. This will be used to provide the Colvars module with the current thermostat target temperature when using a method that needs this information.

3.2 Configuration syntax for the Colvars module

The syntax of the Colvars configuration is "`keyword value`", where the keyword and its value are separated by any white space. The following rules apply:

- keywords are case-insensitive (`upperBoundary` is the same as `upperboundary` and `UPPERBOUNDARY`): their string values are however case-sensitive (e.g. file names);
- a long value or a list of multiple values can be distributed across multiple lines by using curly braces, "{" and "}": the opening brace "{" must occur on the same line as the keyword, following a space character or other white space; the closing brace "}" can be at any position after that;
- many keywords are nested, and are only meaningful within a specific context: for every keyword documented in the following, the "parent" keyword that defines such context is also indicated;
- if a keyword requiring a boolean value (`yes|on|true` or `no|off|false`) is provided without an explicit value, it defaults to '`yes|on|true`'; for example, '`outputAppliedForce`' may be used as shorthand for '`outputAppliedForce on`';
- the hash character # indicates a comment: all text in the same line following this character will be ignored.

The following keywords are available in the global context of the colvars configuration, i.e. they are not nested inside other keywords:

- `colvarsTrajFrequency` ⟨Colvar value trajectory frequency⟩
  Context: global
  Acceptable values: positive integer