Documentation of g_mia and g_mialoc

October 20, 2010

1 What are g_mia and g_mialoc?

g_mia and *g_mialoc* can be used to analyze the morphology of molecular aggregates. The programs accept coordinate and index files in the standard formats supported by the Gromacs software package. For a full description of the method including sample applications you should consult the original paper (M. Fuhrmans, S.J. Marrink. "A tool for the morphological analysis of mixtures of lipids and water in computer simulations". J. Mol. Model., 2010).

1.1 Global morphology

Use *g_mia* to characterize the global structure of the aggregate by computing the Minkowski functionals of the corresponding voxelated image. A file named *mia.xvg* is created, containing a plot of the four Minkowski functionals for the complete input trajectory. Periodic boundary conditions and triclinic boxes are supported.

1.2 Local curvature

Use *g_mialoc* to create images onto which the local values of the Gaussian and mean curvature are mapped as a color code (blue, white and red for negative, zero and positive curvature, respectively). The images are created as two PyMOL files. If you want to use the spatial averaging option with several rotated grid orientations (see below), the area of interest needs to be centered in the box prior to the analysis. With this option, the image will be truncated near the box edges and not take into account periodic boundary conditions.

g_mialoc creates only images for a single snapshot and not the complete trajectory. No video option is implemented at the moment.

This program also creates a *mia.xvg* file containing the global Minkowski functionals. Note, however, that these describe the truncated image and are usually not the information you are looking for.

2 How to compile the source

- Copy the four files *g_mia.c*, *gmx_mia.c*, *g_mialoc.c*, and *gmx_mialoc.c* to the */src/tools/* directory of your Gromacs installation.
- Edit the file *Makefile.am.* Include entries for both mia and mialoc in the same format as the entries for the other Gromacs tools. The exact format will likely depend on the Gromacs version you are using.
- Proceed as usual to compile using automake, e.g. run aclocal, configure, ...
- Create the executables by typing *make mia* and *make mialoc.*

3 User-definable options and parameters

Input files The tool needs a coordinate or trajectory file (*-c*) and an index file (*-n*) in which the particles that correspond to the positive phase are listed.

Imaging options The edge length of the grid (*-dim*), the radius of the spherical cloud used to expand the coordinates (*-sr*) and the number of coordinates generated during the expansion (*-npts*) as well as the minimum number of coordinates mapped onto a grid cell required to count it as positive (*-thresh1*) need to be specified.

As a general consideration, the resolution needs to be high enough to accurately depict the structure to be analyzed, but is limited by memory requirements, due to several three-dimensional arrays required during the computation[1]. In addition, using a high resolution usually requires expansion of the coordinates to avoid artificial empty voxels caused by the limited coordinate-density, which partially offsets the desired high resolution. The radius of the spherical cloud should therefore be chosen as the smallest radius sufficient to avoid noise.

It also turns out that, in order to accurately detect flat morphologies as having zero mean curvature, it is required to calibrate the parameters used. Since molecular aggregates usually have a low short-range order, fluctuations of individual molecules from the mean will show as either bumps or dents in the created image. Since a given resolution does not necessarily have the same propensity for producing bumps as for producing dents a net-curvature will be measured. The threshold parameter can be used to adjust the number of "positive" coordinates mapped onto a single grid-cell required to count that cell as positive to, on average, produce an equal number of bumps and dents and therefore not introduce artificial mean curvature to the measurement.

In addition, it is possible to also use the coordinates of the particles corresponding to the negative phase, mapping them onto the grid as has been described above but counting them as negative. If that is desired, the number of phases to consider must be set from 1 to 2 (*-np*), and the index file needs to contain a second group in which these particles are listed.

If isolated clusters below a certain size are to be removed, the maximal cluster size to be considered noise must be specified (*-cs*).

Averaging options The range over which the local curvatures are averaged over neighboring voxels needs to be specified (*-ar1* and *-ar2*), with a value of zero indicating no averaging. Two values are needed, one for the averaging for every single grid orientation (*-ar1*) and one for the averaging after the values of all grid orientations have been collected (*-ar2*).

If multiple grid orientations are to be used, the number of rotations around every axis (*-nx*, *-ny* and *-nz*) and the corresponding angle increments (*-depsilon*, *-dphi* and *-dtheta*), as well as the radius around the center of the box within which voxels are considered have to be set (*-dr*)[2]. In order to achieve the best result, care must be taken to avoid sampling similar orientations.

In addition, a threshold can be specified to only count voxels as positive if a minimum number of local curvatures corresponding to different rotations has been mapped on that voxel (*-thresh2*). However, unlike the other averaging steps, this option will discard curvature and no longer give the exact results and should therefore be used with care.

[1] no attempt was made to optimize the code in this respect

[2] This radius needs to be specified as a value between 0 and 1 and will be multiplied with half the smallest box dimension internally.