Supplementary Software: Readme

for

"Optimized measurements of separations and angles between intra-molecular fluorescent markers"

by

Kim I. Mortensen, Jongmin Sung,
Henrik Flyvbjerg, and James A. Spudich

September 8, 2015

Python script performing the analysis of diPOLE, i.e. maximum likelihood estimation of location and orientation of a fixed dipole along with the objective's distance from the design focal plane.

The file *diPOLE.py* contains the Python (python.org) code for the calculation of the PSF for fixed dipole emitters imaged close to the design focal plane. The class *dipdistr* contains a function that calculates the exact PSF for given orientation of the dipole. It calls the module *numint.py* that contains some functions for numerical integration. By initiation of the class, an automatic check for already saved normalization parameters is done — if not found, normalization is performed and the parameters are saved to the file *dipolenorm.dat* in the working directory. Any working directory should be used to analyze data from a single wavelength only. Also at initiation, the integrals of the cumulant approximation of the theoretical PSF are calculated for each value of the distance to the design focal plane. Code for the maximum likelihood estimation based on this theoretical PSF is given in the class *MLEwT*. It uses the class *LogLikelihood* to perform the maximum likelihood estimation of the position coordinates, the orientation of the probe, the number of background photons per pixel, the total photon number, and the distance to the design focal plane (see end of code for an example that reads simulated data from the file *data.txt* and estimates the position and orientation of a fixed fluorophore). By using the class *MLEwTcovar*, the code also returns the uncertainties for the estimated parameters.