pm-project-1

Prateek Mehta

October 31, 2012

Contents

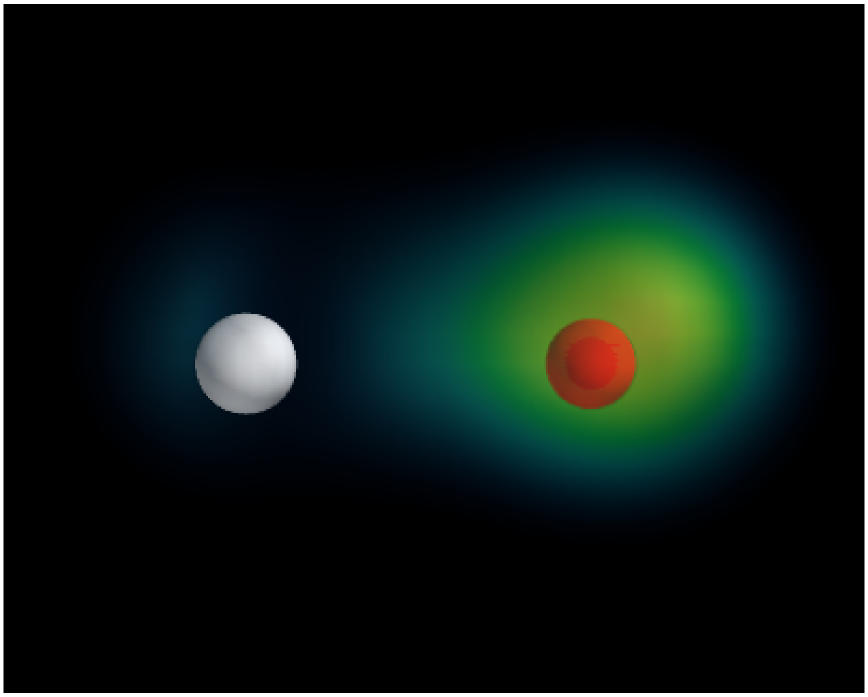
1 Introduction

This project aims to plot electron density as a “fog” rather than with isosurfaces using jasp. The code has been adapted from the examples in dft-book and the example given on http://docs.enthought.com/mayavi/mayavi/auto/example_chemistry.html. The mayavi documentation on volume rendering (http://docs.enthought.com/mayavi/mayavi/mlab.htm volumetric-scalar-data) and the jasp documentation for plotting charge density (https://github.com/jkitchin/jasp/blob/master/CHG.py) have been used as reference. Examples of python code have been written for the CO and the H_2O molecule.

2 CO Example

```
from jasp import *
from enthought.mayavi import mlab
from ase.data import vdw_radii
from ase.data.colors import cpk_colors
from ase import Atom, Atoms
# unit cell parameters
a1=10.
a2=10.
```

```
a3=10.

# creating atoms object
atoms = Atoms([Atom(’C’, [2.422, 0.0, 0.0]),
               Atom(’O’, [3.578, 0.0, 0.0])],
               cell=(a1,a2,a3))
atoms.center()

# running dft calculation
with jasp(’molecules/co-fog’,
          encut=350,
          xc=’PBE’,
          atoms=atoms) as calc:
    calc.calculate()
    atoms = calc.get_atoms()
    x, y, z, cd = calc.get_charge_density()

# getting dimensions of the charge density
n1, n2, n3=cd.shape
mlab.figure(bgcolor=(0, 0, 0))

# plotting the atoms as spheres,

for atom in atoms:
    mlab.points3d(atom.x*n1/a1, # adjusting the atom positions to match shape of cd
                  atom.y*n2/a2,
                  atom.z*n3/a3,
                  scale_factor=(n1/a1)*vdw_radii[atom.number]/5.,
                  resolution=20,
                  # a tuple is required for the color
                  color=tuple(cpk_colors[atom.number]),
                  scale_mode=’none’)

# plotting the charge density
source = mlab.pipeline.scalar_field(cd)
min = cd.min()
max = cd.max()
# vmin and vmax adjusted by iteration
vol = mlab.pipeline.volume(source, vmin=min+.1*(max-min),
                                   vmax=min+(max-min))

# view adjusted by iteration
mlab.view(azimuth=-90, elevation=90, distance=’auto’)

mlab.savefig(’molecules/co-fog.png’)
mlab.show()
```



3 H_2O Example

```
from jasp import *
from enthought.mayavi import mlab
from ase.data import vdw_radii
from ase.data.colors import cpk_colors
from ase import Atom, Atoms

a1=8.
a2=8.
a3=8.

atoms = Atoms([Atom(’H’, [0.5960812,  -0.7677068,   0.0000000]),
               Atom(’O’, [0.0000000,   0.0000000,   0.0000000]),
               Atom(’H’, [0.5960812,   0.7677068,   0.0000000])],
               cell=(a1, a2, a3))
atoms.center()


with jasp(’molecules/water-fog’,
          encut=350,
          xc=’PBE’,
          atoms=atoms) as calc:
    calc.calculate()
    atoms = calc.get_atoms()
    x, y, z, cd = calc.get_charge_density()
```

```


n1, n2, n3=cd.shape
mlab.figure(bgcolor=(0, 0, 0))


for atom in atoms:
    mlab.points3d(atom.x*n1/a1,
                  atom.y*n2/a2,
                  atom.z*n3/a3,
                  scale_factor=(n1/a1)*vdw_radii[atom.number]/5.,
                  resolution=20,
                  color=tuple(cpk_colors[atom.number]),
                  scale_mode='none')


source = mlab.pipeline.scalar_field(cd)
min = cd.min()
max = cd.max()
vol = mlab.pipeline.volume(source, vmin=min+.1*(max-min),
                                   vmax=min+(max-min))

mlab.savefig('molecules/water-fog.png')
mlab.show()
```