

# USER'S GUIDE

## Program for performing DDEC atomic population analysis

### Version 2.1 Beta

October 10, 2012

Please use the most recent version of this program by checking for updates at

<http://sourceforge.net/projects/ddec/>

#### CONTENTS:

- A) [Introduction and Overview](#)
- B) [Summary of the code's capabilities](#)
  - C) [Version Information](#)
  - D) [System Requirements](#)
  - E) [Licensing Information](#)
- F) [Installing and starting the program](#)
- G) [Operating instructions for VASP users](#)
- H) [Operating instructions for Gaussian 09 users](#)
- I) [Operating instructions for GPAW users](#)
- J) [Operating instructions for Siesta users](#)
- K) [Operating instructions for CP2K users](#)
- L) [Reading and visualizing the Chargemol output files](#)
- M) [Calculating the accuracy of the net atomic charges for reproducing the electrostatic potential](#)
- N) [Calculating the accuracy of the atomic spin moments for reproducing the magnetic field](#)
- O) [What to do if you encounter a problem](#)

#### A) Introduction and Overview

The density derived electrostatic and chemical (DDEC) method partitions the electron and spin densities to compute net atomic charges and atomic spin moments. The DDEC net atomic charges are simultaneously optimized to reproduce the chemical states of atoms and the electrostatic potential,  $V(\vec{r})$ , outside the material's electron distribution. DDEC charges are well-suited for studying the chemical properties of materials and for constructing force-fields used in atomistic simulations. The atomic spin moments are simultaneously optimized to reproduce atomic chemical states and the spin contribution to the magnetic field,  $\vec{B}^{\text{spin}}(\vec{r})$ , outside the material's electron distribution. The DDEC method has been designed for accurate analysis of a variety of materials including: (a) molecular systems, (b) porous and non-porous crystalline solids, (c) solid surfaces, (d) nanoclusters, nanotubes, sheets, and other nanostructures. The method can accurately treat covalently and ionically bonded materials. The principal advantages of

the DDEC method are its high accuracy, applicability to a wide range of materials, sound theoretical basis, and basis set independence. Its computational cost scales approximately linearly as the number of atoms in the unit cell increases.

The main criteria I used in constructing the method and associated software program is to focus on accuracy, scaling efficiency, and robustness. I placed less emphasis on producing the fastest running code, but hope to address this issue in future versions. Consequently, you should find the code to be very accurate and robust, but you will need to allow many hours for DDEC population analysis to complete. With the current code, it takes approximately one cpu week to perform DDEC analysis on a periodic system containing 500 atoms in the unit cell. As such, this code is primarily intended for running on computational clusters rather than on your desktop computer. If your goal is to compute net atomic charges in just a couple of minutes (“instantly”), this code will probably not meet your needs. If your goal is to compute the most accurate net atomic charges and atomic spin moments and you do not mind waiting a few days for the results, this code should meet your needs.

More information can be found in

**This code uses the DDEC/c3 charge partitioning method described in:** T. A. Manz and D. S. Sholl, “Improved Atoms-in-Molecule Charge Partitioning Functional for Simultaneously Reproducing the Electrostatic Potential and Chemical States in Periodic and Non-Periodic Materials,” *J. Chem. Theory Comput.* **8** (2012) 2844-2867.

**This code uses the spin partitioning method described in:** T. A. Manz and D. S. Sholl, “Methods for Computing Accurate Atomic Spin Moments for Collinear and Noncollinear Magnetism in Periodic and Nonperiodic Materials,” *J. Chem. Theory Comput.* **7** (2011) 4146-4164.

T. A. Manz and D. S. Sholl, “Chemically Meaningful Atomic Charges that Reproduce the Electrostatic Potential in Periodic and Nonperiodic Materials,” *J. Chem. Theory Comput.* **6** (2010) 2455-2468.

Although the code accepts a variety of input file types (i.e., VASP, wfx, cube, and xsf), it has been most often used with VASP and .wfx files. For periodic systems, a computationally efficient and accurate strategy is to use the projector augmented wave (PAW) method in VASP. To date, the code has been used in this manner to analyze approximately 1000 periodic materials with unit cells ranging in size from one atom to more than 600 atoms. Both the VASP and DDEC codes are extremely reliable, so you can reasonably expect this approach to perform flawlessly and yield great results. In principle, SIESTA output can be converted to an xsf file and analyzed using the DDEC program; however, this conversion is somewhat tedious and less often used. For nonperiodic (e.g., molecular) systems, the preferred strategy is to generate a .wfx file using the latest Gaussian 09 release. The .wfx file was specifically designed to contain the appropriate information for performing atom-in-molecule population analysis.<sup>1</sup> The .wfx file contains the gaussian basis set specifications, nuclear positions, total net charge, basis set coefficients for the natural orbitals, natural orbital occupancies, and core electron densities for pseudo-potentials. In my experience, essentially all of the problems encountered when using a .wfx file have to do with it being generated improperly. The reliability of generating the .wfx file has improved between Gaussian 09 releases, so it is best to use the latest Gaussian 09 release for generating this file. For other types of quantum chemistry software (e.g., GPAW and CP2K), the DDEC program can accept input in the form of cube files containing the valence and spin densities.

---

<sup>1</sup> <http://aim.tkgristmill.com/wfxformat.html>

## B) Summary of the code's capabilities

1. Computes DDEC net atomic charges, dipoles, and quadrupoles. Also computes the effective decay exponent for each atom's electron density tail.
2. Computes DDEC atomic spin moments for collinear and non-collinear magnetic systems.
3. Reads VASP files, wfx files, cube files, and xsf files. This allows compatibility with a number of different computational chemistry programs including but not limited to VASP, Gaussian, SIESTA, GPAW, and CP2K.
4. Compatible with electron density files produced from any quantum chemistry method including but not limited to density functional theory, coupled cluster theory, configuration interaction methods, and multi-reference methods like CAS-SCF.
5. Compatible with different strategies for treating core electrons during the *ab initio* calculation: (a) all-electron relaxed core, (b) all-electron frozen core (e.g., projector augmented wave method), and (c) effective core potential or pseudo-potential. In case (c), the DDEC program includes the core electron density replaced by the pseudo-potential using core electron reference densities.
6. Performs numerous integration checks to make sure all electrons are properly accounted for. It makes sure the number of valence and core electrons integrate to the expected values, and it makes sure the grid spacing is adequate.
7. Reference densities are currently available for the following 74 elements (listed in order of atomic number): H, He, Li, Be, B, C, N, O, F, Ne, Na, Mg, Al, Si, P, S, Cl, Ar, K, Ca, Sc, Ti, Cr, Mn, Fe, Co, Ni, Cu, Zn, Ga, Ge, As, Se, Br, Kr, Rb, Sr, Y, Zr, Mo, Ru, Rh, Pd, Ag, Cd, In, Sn, Sb, Te, I, Xe, Cs, Ba, Ce, Nd, Sm, Eu, Gd, Ho, Er, Hf, Os, Ir, Pt, Au, Hg, Tl, Pb, Bi, At, Rn, Fr, Ra, and U. If you need a reference density that is not yet available, the program will terminate with a message like 'The atomic density file listed above does not exist.'. If this occurs, please contact the email address given at the end of this document.
8. Computes the relative root mean squared error (RRMSE) between the electrostatic potential of a point charge model and the electrostatic potential  $V(\vec{r})$  of the *ab initio* electron density,  $\rho(\vec{r})$ . To compute the RRMSE,  $V(\vec{r})$  must be in a Gaussian-style cube file (non-periodic system) or a VASP LOCPOT file (periodic system).
9. Computes the relative mean absolute error (RMAE) between the magnetic field of an atomic spin moment model and the spin-derived magnetic field,  $\vec{B}^{\text{spin}}(\vec{r})$ , of the *ab initio* spin density.
10. For non-collinear magnetic systems, two descriptors,  $\Xi^{\text{mag}}$  and  $\Xi^{\text{dir}}$ , are automatically computed to assess the appropriateness of the exchange-correlation functional. More information on these descriptors can be found in the publication listed in section A above.

## C) Version Information

The main differences between version 2.1 and the prior code version (2.0) are:

1. Spin partitioning is approximately five times faster by using a function lookup table to avoid unnecessary re-computation of function values. The values of the computed atomic spin moments are not affected.
2. Reference densities have been added for 24 new elements.
3. The error in the integrated electron count is now printed to the DDEC\_net\_atomic\_charges.xyz file. In addition, the integral cutoffs when using .wfx file input have been made tighter. These changes are quantify and improve the computational precision.

## D) System Requirements:

1. To run this software, you need to have
  - i) Matlab version 7 or later. (Program does not run with Matlab 6 or earlier.)
  - ii) One of the following computational chemistry programs to generate the electron density files: (a) VASP version 4.5 or later, (b) Gaussian 09 revision B.01 or later, (c) a recent version of SIESTA, or (d) a program (e.g., CP2K or GPAW) that can generate cube files.
  - iii) If you are using VASP software to generate the electron density files, you should use the projector augmented wave (PAW) method not the ultrasoft pseudopotentials, because the PAW method is intrinsically more accurate than pseudo-potential methods.
  - iv) Approximately 2 GB of RAM is needed for calculating charges in systems containing ca. 100 atoms. More memory may be required for analyzing larger systems.
  - v) Limitations on the size of Matlab arrays depend on the Matlab version and operating system. Recent 64 bit versions of Matlab allow arrays of almost unlimited size, while earlier 32 bit versions limit array size. If you are analyzing systems containing hundreds or thousands of atoms, a recent 64 bit version may be required.

The program used to generate the electron density files can be installed on either the same or a different computer as the one running Matlab. The DDEC program is designed to work on a variety of operating systems. It has been tested on linux and windows machines.

2. If you are a regular user of a computational chemistry program whose density files cannot currently be read by the Chargemol program, and you have an interest in helping to upgrade the Chargemol program to read those files, please feel free to contact us via the email listed at the end of this document.

## E) Licensing information:

This software is distributed under the license described below, which is compliant with the Open Source Initiative found at <http://www.opensource.org/docs/osd>. You are not licensed to use or modify this software unless you agree to all of the licensing terms below:

1. Derived Works and Integrity of The Author's Source Code: Derived works must be distributed under exactly the same terms as this license. **The source code is restricted from being distributed in modified form; however, this license allows the distribution of "patch files" with the source code for the purpose of modifying the program at build time.**
2. Free Redistribution: The license does not restrict any party from selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources. The license does not require a royalty or other fee for such a sale. This license does not restrict other software that may be distributed along with the licensed software.
3. Source Code: The program includes source code, and allows distribution in source code as well as compiled form. (Deliberately obfuscated source code and intermediate forms such as the output of a preprocessor or translator are not allowed.)
4. Distribution of License: The rights attached to the program apply to all to whom the program is redistributed without the need for execution of an additional license by those parties.

5. Scientific publications that use computational results of the program are required to properly cite the original research papers describing the methods used in the program, according to best scientific practice.

*Note:* For the current software version, this would include citation of the following papers:

(a) for the calculation of atomic charges, please cite: (i) T. A. Manz and D. S. Sholl, "Improved Atoms-in-Molecule Charge Partitioning Functional for Simultaneously Reproducing the Electrostatic Potential and Chemical States in Periodic and Non-Periodic Materials," *J. Chem. Theory Comput.* **8** (2012) 2844-2867.

(ii) T. A. Manz and D. S. Sholl, "Chemically Meaningful Atomic Charges that Reproduce the Electrostatic Potential in Periodic and Nonperiodic Materials," *J. Chem. Theory Comput.* **6** (2010) 2455-2468.

(b) for the calculation of RMSE and RRMSE, please cite: (i) T. A. Manz and D. S. Sholl, "Improved Atoms-in-Molecule Charge Partitioning Functional for Simultaneously Reproducing the Electrostatic Potential and Chemical States in Periodic and Non-Periodic Materials," *J. Chem. Theory Comput.* **8** (2012) 2844-2867. (ii) T. Watanabe, T. A. Manz, and D. S. Sholl, "Accurate Treatment of Electrostatics during Molecular Adsorption in Nanoporous Crystals without Assigning Point Charges to Framework Atoms," *J. Phys. Chem. C*, **115** (2011) 4824 – 4836.

(c) for the calculation of atomic spin moments, please cite: T. A. Manz and D. S. Sholl, "Methods for Computing Accurate Atomic Spin Moments for Collinear and Noncollinear Magnetism in Periodic and Nonperiodic Materials," *J. Chem. Theory Comput.* **7** (2011) 4146-4164.

6. This license makes no discrimination against persons or groups or fields of endeavor. The rights attached to this program do not depend on the program's being part of a particular software distribution. No provision of this license is predicated on any particular style of interface.

#### **F) Installing and starting the program:**

- 1) Download and unzip the file **chargemol\_10\_10\_2012.zip** to create the folder **chargemol\_10\_10\_2012** (or whatever else you want to call it) containing the subfolders **sourcecode**, **examples**, and **atomic\_densities**, and the files **chargemol\_job.m**, **instructions\_10\_10\_2012.pdf**, and **example\_script.sh**.
- 2) Move the folder **chargemol\_10\_10\_2012** to the desired location on your hard drive or home directory. Note that when moving files between Windows and Linux systems, it is important to indicate whether the files are text (ASCII) or binary, because text files are stored in slightly different formats on these two operating systems. All of the files in the unzipped **chargemol\_10\_10\_2012** folder except **instructions\_10\_10\_2012.pdf** are in text (ASCII) format.
- 3) Create a separate folder for each system that you want to compute DDEC charges for. As explained in the following sections, you will need to place certain files in each of these folders containing information about the electron density and other system properties. Copy the **chargemol\_job.m** file into each of these folders. If you are copying from a Windows to a Linux machine, make sure that you transfer the file in text (ASCII) format.
- 4) Open the file **chargemol\_job.m** and change the **sourcecode\_directory** to the appropriate value. Also, make sure the other variables in the **chargemol\_job.m** file are set to appropriate values.

- 5) If you want to run the job directly, start Matlab, change to the directory containing the job's files, and type `chargemol_job` at the Matlab command prompt. This should cause the Chargemol program to run, and the program will create a `DDEC_net_atomic_charges.xyz` file. If you want to submit the job to a queue, you will need to create a script file. An example script file is `example_script.sh`, which would be submitted by typing `qsub example_script.sh` at the linux command prompt. The details of how to prepare the script file vary from system to system. Please contact an experienced person in your research group if you have any questions about how to create and submit a script file on the system you are using.

#### G) Operating instructions for VASP users:

- 1) Set up a POSCAR file with the final set of atomic positions: Before generating the density files used by the Chargemol program, you must first have the final set of atomic positions. If you want to optimize the atomic positions in the system you are studying, you must do this prior to attempting to generate the density files used by the Chargemol program. Once you have the final set of atomic positions, place them in a `POSCAR` file in the usual manner.
- 2) Set up the INCAR file to compute the needed density files: Specify `NSW = 0` (this tells VASP not to change the geometry) and `LCHARG = .TRUE.` and `LAECHG = .TRUE.` in the `INCAR` file (this tells VASP to generate the needed electron density files). We want to re-iterate that NSW must be set to zero in this calculation. Specify `PREC = Accurate` in the `INCAR` file to generate an appropriate grid spacing for calculating the DDEC charges. The other parameters in the `INCAR` file are specific to your particular system and should be specified as described in the VASP manual.
- 3) Set up the KPOINTS file: In the `KPOINTS` file, you need to specify the number of k-points along each of the three lattice directions. We recommend that the number of k-points along a lattice direction times the length of the corresponding lattice vector be approximately 16 Å or greater. For example, if the lengths of the lattice vectors for your unit cell are 3 Å, 9 Å, and 23 Å, you might want to use 6 k-points along the first lattice vector, 2 k-points along the second lattice vector, and 1 k-point along the third lattice vector. You can specify more k-points if you wish, but this probably will not change the values of the computed charges.
- 4) Set up the POTCAR file: The `POTCAR` file is set up in the usual manner using projector augmented wave (PAW) basis sets. For the current version, you cannot use ultrasoft pseudopotentials to generate density files needed for the Chargemol program. This is not a limitation of the Chargemol program. Rather, I've consciously chosen not to support the less reliable pseudo-potential method for codes that implement the more accurate projector augmented wave (PAW) method.
- 5) Run the VASP program to produce the `AECCAR0`, `AECCAR2`, and `CHG` files, along with other files. The `AECCAR0` file holds the core electron density while the `AECCAR2` file holds the valence electron density. The `CHG` file holds the valence pseudo-density and the spin densities (for spin-polarized calculations). `AECCAR0`, `AECCAR2`, `CHG`, and `POTCAR` are the only files needed for the subsequent charge analysis.
- 6) Run the Chargemol program to compute the DDEC charges: Place the `chargemol_job.m` file in the same directory as the `AECCAR0`, `AECCAR2`, `CHG`, and `POTCAR` files. Specify `input_type = 2` (VASP), `periodicA = 1`, `periodicB = 1`, and `periodicC = 1`, and the net charge of the unit cell (`netcharge = __`, fill in number) in the `chargemol_job.m` file. Make sure VASP version 4 or 5 is set correctly in the `chargemol_job.m` file. Run this file

either by typing **chargemol\_job** at the Matlab command prompt or else by submitting a batch script placed in this same directory. (Details of the batch script vary from system to system. See **example\_script.sh** for an example.)

- 7) Special instructions for VASP non-collinear magnetism calculations: Instructions for how to perform a non-collinear magnetism calculation in VASP using the keyword `Lnoncollinear = .true.` are described in the VASP manual. To generate the necessary charge files, be sure to also specify `LCHARG = .true.` and `LAECHG = .true.` as described above. If desired, spin orbit coupling can be included as described in the VASP manual. After the non-collinear magnetism in VASP completes, change the name of the CHG file to **CHG\_noncollinear**. (This is how the Chargemol program is alerted that the information contained in the file is for noncollinear magnetism.) Run the Chargemol program as described above using **CHG\_noncollinear** in place of the CHG file. *Note:* If you set SAXIS in the **INCAR** file, be sure set SAXIS in the **chargemol\_job.m** file to this same value. (For example, add a line like `SAXIS = [0.1, -0.1, 0.9]` to the change constants section of the **chargemol\_job.m** file.) If you did not set SAXIS in the **INCAR** file, then don't set it in the **chargemol\_job.m** file either.

#### H) Operating instructions for Gaussian 09 users:

1) Generate the **.wfx** file using Gaussian 09 Revision B.01 or later. Set up and run your Gaussian job in the usual manner. To generate the **.wfx** file, include the following keywords in route line of the Gaussian 09 input file:

for spin restricted calculations: `density = current output = wfx pop=NO`

for spin polarized calculations: `density = current output = wfx pop=NOAB`

The name of the **.wfx** file must be specified at the bottom of the Gaussian 09 input file.

2) Run the Chargemol program to compute the DDEC charges: Place the **chargemol\_job.m** file in the same directory as the **.wfx** file. Specify `input_type = 5` and the **.wfx** filename in the **chargemol\_job.m** file. Run this file either by typing **chargemol\_job** at the Matlab command prompt or else by submitting a batch script placed in this same directory. (Details of the batch script vary from system to system. See **example\_script.sh** for an example.)

If you run into issues generating or using **.wfx** files, please contact the email listed at the end of this document.

#### I) Operating instructions for GPAW users:

- 1) Generate a cube file named **total\_density.cube** with a grid spacing of  $\sim 0.07 \text{ \AA}$  (0.14 bohr) along each lattice direction. This file should contain the total electron density, not the valence density. The dimensions of the cube file must correspond to the unit cell. To compute the atomic spin moments for spin-polarized (magnetic) systems, you will also need to generate a similar cube file named **spin\_density.cube** that contains the spin density.
- 2) Run the Chargemol program to compute the DDEC charges: Place the **chargemol\_job.m** file in the same directory as the **valence\_density.cube** file (and the **spin\_density.cube** file for magnetic systems). Specify `input_type = 6` and the system's net charge (`netcharge = __`, fill in number) in the **chargemol\_job.m** file. You will also need to specify `periodicA=1`, `periodicB=1`, and `periodicC=1` in this file. Run this file either by typing



**chargemol\_job** at the Matlab command prompt or else by submitting a batch script placed in this same directory. (Details of the batch script vary from system to system. See **example\_script.sh** for an example.)

#### J) Operating instructions for Siesta users:

- 1) For the computation of net atomic charges, basis sets of DZP (double zeta with polarization) or higher quality (e.g., TZP or TZDP) are recommended. Basis sets generated without a confinement potential may experience an abrupt change in value at the cutoff radius which spoils the accuracy of geometry optimizations. Basis sets prepared using a confinement potential are strongly recommended, because these basis sets decay smoothly to zero at the cutoff radius producing accurate forces and geometries. For this reason, all Siesta calculations should use basis sets generated with a confinement potential.
- 2) A MeshCutoff value of 300 Ry is recommended. Lower MeshCutoff values give grids that are too coarse while higher MeshCutoff values give finer grids than necessary. According to tests we performed, FilterCutoff does not need to be specified in the Siesta input file to get accurate results when using **MeshCutoff = 300 Ry**.
- 3) Include the keyword **SaveRho = .true.** in the Siesta input file and run the Siesta calculation in the usual manner to generate the **SystemLabel.RHO** file that contains the valence charge density.
- 4) Download the Sies2xsf program written by Andrei Postnikov from <http://www.home.uni-osnabrueck.de/apostnik/download.html>. (Sies2xsf converts **SystemLabel.RHO** into an .xsf formatted file readable by the XCrySDens and Chgarmol programs). After downloading and uncompressing Sies2xsf, open the file Rho2xsf and uncomment the lines shown below by removing the C character at the line's beginning. This turns on the option for writing a periodic unit cell to the .xsf file.

(Remove C from each line shown below.)

```
C      write (6,702)
C 101  write (6,703,advance="no")
C      read (5,*) labbox
C      if (labbox.eq.'Y'.or.labbox.eq.'y') then
C          write (iol,'(a7)') 'CRYSTAL'
C          write (iol,'(a7)') 'PRIMVEC'
C          write (iol,201) (rbox(ii,1),ii=1,3)
C          write (iol,201) (rbox(ii,2),ii=1,3)
C          write (iol,201) (rbox(ii,3),ii=1,3)
C          write (iol,'(a9)') 'PRIMCOORD'
C          write (iol,'(i5,i2)') nbox,1
C      else if (labbox.eq.'N'.or.labbox.eq.'n') then
C          write (iol,'(a5)') 'ATOMS'
C      else
C          write (6,*) "I do not understand:"
C          goto 101
C      endif
```

**Do not uncomment any other lines in the Rho2xsf file!**

After uncommenting these lines, compile the Sies2xsf program. If you have problems installing or running Sies2xsf, please contact Andrei Postnikov or ask your question to the Siesta mailing list. I am unable to answer technical questions about the Sies2xsf program.



- 5) Run the Sies2xsf program and answer questions indicated. The Sies2xsf program will ask for an origin, grid vectors, and number of points along each grid vector. The Chargemol program requires at least one grid point per 0.2 bohr (0.1 Å), so it is wise to choose enough points to give a spacing of 0.1 to 0.15 bohr along each grid vector. For periodic systems, the grid vectors should be chosen equal to the unit cell's lattice vectors with the origin chosen as (0.0, 0.0, 0.0). For non-periodic systems, the origin and grid vectors should be chosen to place the molecule in the center of cell, so that the distance between the box's surface and each atom is at least 4 Å. IMPORTANT: The Sies2xsf program will indicate the number of atoms found to be located inside the box. If this number is the same as the number of atoms in your system, then you're all set. If this number is less than the number of atoms in your system, then choose a very tiny increase in the box's size by choosing the origin to be (-0.00000001, -0.00000001, -0.00000001) or something similar until you get the correct number of atoms in the box. (This may be necessary, for example, if the nucleus for one atom is located precisely at the box's edge so that a tiny difference in the grid vector will place the atom either inside or outside the box.) IMPORTANT: The difference between the grid vectors and the lattice vectors must be smaller than 0.000001 bohr or else the Chargemol program will complain. The Sies2xsf program will ask for a 3D property to add. Type **RHO** and then hit enter. The Sies2xsf will say it has added the density (RHO) to the xsf file. If available, the spin density is automatically included in the RHO information, so you don't need to add any other 3D properties. When Sies2xsf asks if you wish to add yet another 3D property, hit enter to indicate you are finished adding the desired properties.
- 6) Use an editor to open the .xsf generated by Sies2xsf. If your system is non-periodic the first line of the file should say ATOMS. If your system is periodic, the first line of the file should say CRYSTAL and the second line should say PRIMVEC. You cannot run a charge calculation for a periodic system using the non-periodic ATOMS notation. Therefore, if your system is supposed to be periodic but the first line of the .xsf file says ATOMS, you will need to regenerate the .xsf file. To produce proper .xsf files for periodic systems, please see the instructions above for uncommenting the appropriate lines of rho2xsf and recompiling Sies2xsf.
- 7) After generating the .xsf file, open the **chargemol\_job.m** file and change the value of the variable `xsf_inputfile=' valence_density.xsf '` to the appropriate filename. Also set `input_type=3`, specify the appropriate value for `net_charge`. Make sure the variables `periodicA`, `periodicB`, and `periodicC` are set the appropriate values (1 for periodic, 0 for non-periodic). Make sure the `atomic_densities_directory` and `sourcecode_directory` are set to the correct values.
- 8) To run properly the Chargemol program must know the number of core electrons for each element. By default, these values are set equal to largest noble gas core. For example, Ru (atomic number 44) would have `num_core(44) = 36` by default since Kr (atomic number 36) is the largest noble gas with atomic number less than 44. As a second example, Kr corresponds to a default value `num_core(36) = 18`, since Ar (atomic number 18) is the noble gas appearing just before Kr. The number of core electrons for each element can be changed by placing an appropriate entry in the **chargemol\_job.m** file. For example, if a pseudo-potential for Ag (atomic number 47) used a core of 28 electrons ([Ar] + the 3d electrons), this could be specified by placing `num_core(47) = 28` in the **chargemol\_job.m** file. Other possible values include `num_core(47) = 36` (the default), `num_core(47) = 30` (4p electrons moved to the valence space), etc. Use the following steps to determine the number of core electrons from a Siesta output file:
  - (a) Open the Siesta output file and locate the following line for each element used in the calculation:

Vna: chval, zval: 4.000 4.000

The atomic number minus zval is the number of core electrons. The element for this example was Si (atomic number 14) with zval = 4, so  $\text{num\_core}(14) = 14 - 4 = 10$ . Since this is the default value used in the Chargemol program (for Si the largest noble gas core is [Ar] with ten electrons), nothing needs to be done. If instead zval was 10.00, then  $\text{num\_core}(14) = 4$  would need to be specified in the `chargemol_job.m` file.

- (b) For a given pseudo-potential, you only need to do step (a) once for each element. Moreover, it doesn't hurt to have `num_core` specified for extra elements in the `chargemol_job.m` file. If you repetitively use the same pseudo-potentials, you can create a `chargemol_job.m` file with `num_core` values for all elements you commonly study and re-use this same `chargemol_job.m` file to compute the net charges for your different Siesta jobs.
  - (c) If you mess up step (a), the Chargemol program will probably catch your mistake and terminate with a warning *and* the number of valence electrons does not sum correctly. If the program doesn't complain *and* the computed net atomic charges are reasonable, then you can be sure the number of core electrons was correctly specified.
- 9) After making sure everything is set up correctly in the `chargemol_job.m` file, place it in the same directory as the `.xsf` file. Run the calculation by typing `chargemol_job` at the Matlab prompt, or else by submitting a batch script placed in this same directory. (Details of the batch script vary from system to system. See `example_script.sh` for an example.)
- 10) Special instructions for SIESTA non-collinear magnetism calculations: You only need to make sure the appropriate type of spin computation is performed in Siesta. The Chargemol program will automatically detect what type of spin information is available and perform the corresponding analysis. For spin restricted calculations, the Chargemol program will automatically compute just the net atomic charges. For collinear magnetism, the Chargemol program will automatically compute the net atomic charges and collinear spin moments. For non-collinear magnetism, the Chargemol program will automatically compute the net atomic charges and non-collinear spin moments. Extensive tests have not been performed for non-collinear output from Siesta, so if you encounter a problem please contact me via the email at the end of this document.

#### K) Operating instructions for CP2K users:

- 1) Generate a gaussian-style cube file named `valence_density.cube` with a grid spacing of 0.1 Å (0.2 bohr) or finer along each lattice direction. The dimensions of the cube file must correspond to the unit cell. To compute the atomic spin moments for spin-polarized (magnetic) systems, you will also need to generate a similar cube file named `spin_density.cube` that contains the spin density.
- 2) To run properly the Chargemol program must know the number of core electrons for each element. By default, these values are set equal to largest noble gas core. For example, Ru (atomic number 44) would have  $\text{num\_core}(44) = 36$  by default since Kr (atomic number 36) is the largest noble gas with atomic number less than 44. As a second example, Kr corresponds to a default value  $\text{num\_core}(36) = 18$ , since Ar (atomic number 18) is the noble gas appearing just before Kr. The number of core electrons for each element can be changed by

placing an appropriate entry in the `chargemol_job.m` file. For example, if a pseudo-potential for Ag (atomic number 47) used a core of 28 electrons ([Ar] + the 3d electrons), this could be specified by placing `num_core(47) = 28` in the `chargemol_job.m` file. Other possible values include `num_core(47) = 36` (the default), `num_core(47) = 30` (4p electrons moved to the valence space), etc. If you are unsure how many core electrons to specify, check the details of the pseudo-potential you are using.

- 3) For a given pseudo-potential, you only need to do step (2) once for each element. Moreover, it doesn't hurt to have `num_core` specified for extra elements in the `chargemol_job.m` file. If you repetitively use the same pseudo-potentials, you can create a `chargemol_job.m` file with `num_core` values for all elements you commonly study and re-use this same `chargemol_job.m` file to compute the net charges for your different CP2K jobs. If you mess up step (2), the Chargemol program will probably catch your mistake and terminate with a warning that the number of valence electrons does not sum correctly. If the program doesn't complain and the computed net atomic charges are reasonable, then you can be sure the number of core electrons was correctly specified.
- 4) Run the Chargemol program to compute the DDEC charges: Place the `chargemol_job.m` file in the same directory as the `valence_density.cube` file (and the `spin_density.cube` file for magnetic systems). Specify `input_type = 4` and the system's net charge (`netcharge = __`, fill in number) in the `chargemol_job.m` file. You will also need to specify `periodicA=1`, `periodicB=1`, and `periodicC=1` in this file. Run this file either by typing `chargemol_job` at the Matlab command prompt or else by submitting a batch script placed in this same directory. (Details of the batch script vary from system to system. See `example_script.sh` for an example.)

#### L) Reading and visualizing the Chargemol output files:

- 1) The `DDEC_net_atomic_charges.xyz` file: This file contains the atomic coordinates and partial atomic charges in a format that can be read and displayed by the Jmol program. An example is shown below:

```
8 <-- number of atoms in system or unit cell
the system's unit cell information is contained on this line
N 1.588924 -0.139813 -2.223765 -0.858378
N 1.571288 0.130784 2.236801 -0.853561
N -1.573173 2.234910 -0.141791 -0.855478
N -1.587036 -2.225883 0.128751 -0.854863
B -1.241087 0.109136 1.735884 0.856344
B 1.239611 1.737448 -0.100494 0.856408
B -1.227320 -0.102091 -1.746056 0.853826
B 1.228792 -1.744492 0.110672 0.855703
```

The lines after the comment line list information for each of the atoms. The first column is the atomic symbol. Columns two, three, and four contain the X, Y, and Z coordinates, respectively. The net atomic charges (shown in red above) are found in the fifth column. Additional output including the atomic dipoles and quadrupoles, and a fit of the valence density of each atom to a spherically symmetric exponentially decaying function, are displayed in a section below this in the `DDEC_net_atomic_charges.xyz` file.

- 2) For magnetic systems, the atomic spin moments of each atom are contained in the file `DDEC_atomic_spin_moments.xyz`, which has the following format for a collinear magnetic system:

```
2
jmolscript: load "" {1 1 1} spacegroup "x,y,z" unitcell [{ 10.000000 0.000000 0.000000 }, {
0.000000 10.000000 0.000000 }, { 0.000000 0.000000 10.000000 }]
Mg 5.000000 5.000000 5.000000 0.942040
I 5.000000 5.000000 7.609670 0.052816

Collinear spin population analysis was performed
The total spin magnetic moment of the unit cell is 0.994856
```

Chargemol version 1.4.1 Beta released on 03/17/2011  
See [ddec.sourceforge.net](http://ddec.sourceforge.net) for latest version.

```
Computational parameters:
spin_ref_fraction = 0.50
Number of radial integration shells = 75
Cutoff radius (pm) = 400
convergence tolerance = 0.000050
Number of iterations to convergence = 10
The higher multipole contribution to Bspin per atom = 3.633840e-01
The relative higher multipole contribution to Bspin = 7.034835e-01
```

26-May-2011 10:55:30

The format of the **DDEC\_atomic\_spin\_moments.xyz** file is similar to that of the **DDEC\_net\_atomic\_charges.xyz**, with the atomic spin moments displayed in the fifth column. A positive value indicates spin up magnetism, and a negative value indicates spin down magnetism.

For a non-collinear magnetic system, the **DDEC\_atomic\_spin\_moments.xyz** file has the following format:

```
2
jmolscript: load "" {1 1 1} spacegroup "x,y,z" unitcell [{ 10.000000 0.000000 0.000000 }, {
0.000000 10.000000 0.000000 }, { 0.000000 0.000000 10.000000 }]
Mg 5.000000 5.000000 5.000000 0.904210 0.521873 0.522044 0.522221
I 5.000000 5.000000 7.609670 0.053736 0.031018 0.031023 0.031033
```

```
Noncollinear spin population analysis was performed
The total spin magnetic moment vector of the unit cell is 0.552891 0.553066 0.553254
```

Chargemol version 1.4.1 Beta released on 03/17/2011  
See [ddec.sourceforge.net](http://ddec.sourceforge.net) for latest version.

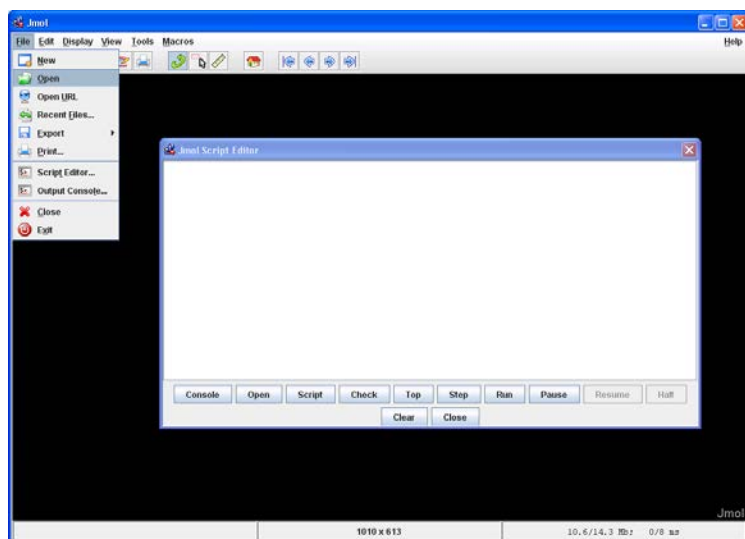
```
Computational parameters:
spin_ref_fraction = 0.50
Number of radial integration shells = 50
Cutoff radius (pm) = 300
convergence tolerance = 0.000050
Number of iterations to convergence = 7
The higher multipole contribution to Bspin per atom = 3.695516e-01
The relative higher multipole contribution to Bspin = 7.156908e-01
```

11-Jun-2011 18:59:58

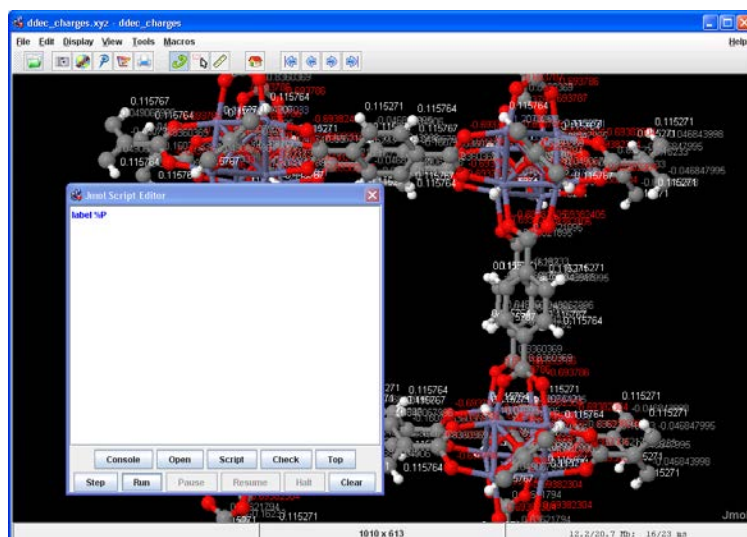
This file is also readable by Jmol. In this case, the fifth column (e.g. 0.904210) contains the magnitude of the spin moment vector for each atom, while the last three columns contain the vector's x, y, z components. The spin moments are listed in as number of electrons. For example, a hydrogen atom whose electronic spin moment is aligned along the z-direction would have a spin moment 0.0 0.0 1.0.

### 3) Using Jmol to visualize the output files:

- download the latest version of Jmol from [jmol.sourceforge.net](http://jmol.sourceforge.net)  
(Jmol version 12 or higher is required to load the periodic unit cell information.)
- After starting Jmol, use the File/Open menu to browse to the directory containing the **ddec\_charges.xyz** file and open it. If the Jmol Script Editor is not open, you can open it using the File/Script Editor menu command.



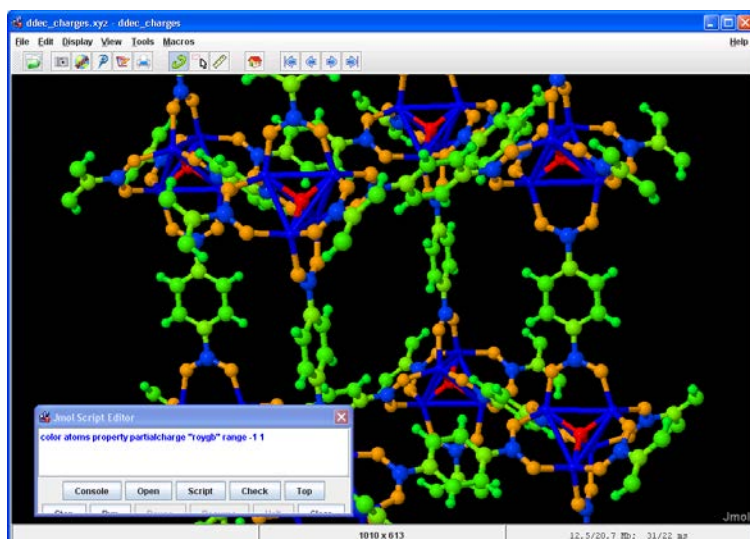
- c) To view the net atomic charges next to each atom, type label %P in the Jmol Script Editor window and then hit Run:



- d) Atoms can be colored according to charge by typing any one of the following commands in the Jmol Script Editor:

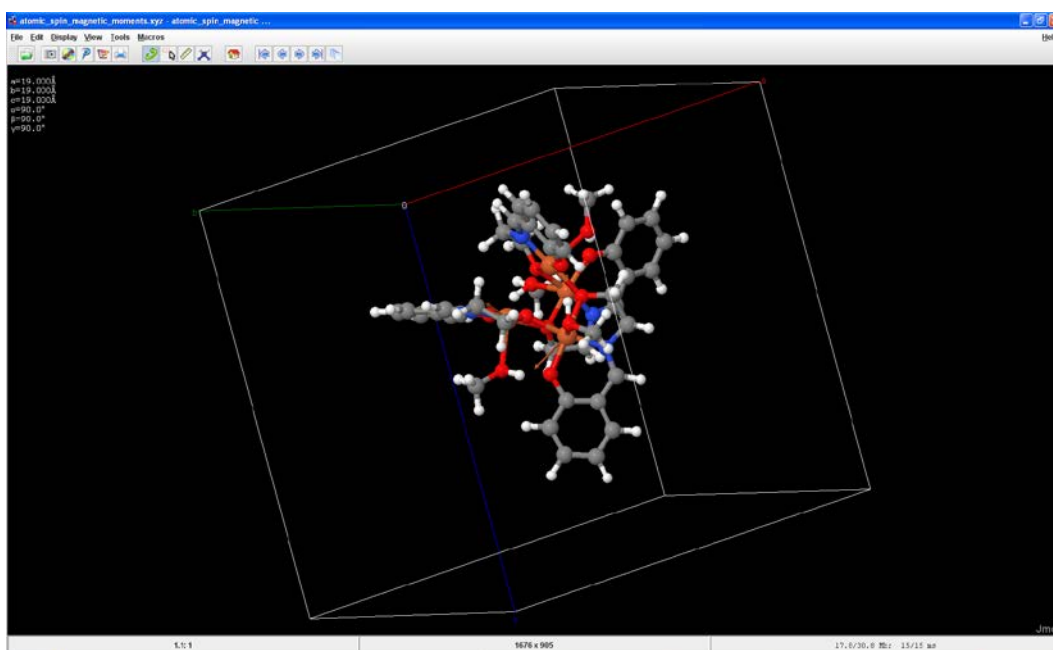
color partialcharge  
 color atoms property partialcharge  
 color atoms property partialcharge "rwb" range -1 1  
 color atoms property partialcharge "roygb" range -1 1

(You can change the range values to get different colors displayed.)



For magnetic systems, the spin magnetic moments of each atom are visualized in the same way, except the [atomic\\_spin\\_magnetic\\_moments.xyz](#) file must be opened. The Jmol commands label %P and color partialcharge display the numeric and color values of the spin moments, respectively. For collinear systems, the value will be either positive (spin up) or negative (spin down).

For non-collinear systems, the scalar value is positive and indicates the magnitude of the spin vector. By default, Jmol animates the vectors as vibrations. Go to Tools > Vibrate > Stop vibration to make the atoms stop moving. Once this is done, the spin moment vectors will be automatically displayed for each atom. Below is an example with spin moment vectors displayed. (They are negligible on all atoms except the orange ones.)



### M) Calculating the accuracy of the net atomic charges for reproducing the electrostatic potential:

The accuracy of a point charge model for reproducing the ab initio electrostatic potential energy surface can be quantified by the root-mean-square error (RMSE),

$$RMSE = \left\{ \sum_i [V_i^q + V_{offset} - V_i^{DFT}]^2 / N_{points} \right\}^{1/2}, \quad (1)$$

where  $V_i^q$  ( $V_i^{DFT}$ ) is the potential energy at grid point  $i$  due to the point charges (DFT EPES), and  $V_{offset}$  equals the average of  $V_i^{DFT} - V_i^q$  over the entire set of valid grid points.  $V_{offset}$  is re-computed for each point charge model and for the case when all charges are set to zero. The sum in this expression is taken over the set of all valid grid points. An inner and outer van der Waals (vdW) multiplier are used to describe the volume of valid grid points. Points closer to any atom than the inner multiplier times its vdW radius are excluded. Likewise, points that are farther than the outer multiplier times the vdW radius of every atom are excluded. That is, valid grid points lie between the surfaces described by inner multiplier x vdW radii and outer multiplier x vdW radii. The relative root mean squared error (RRMSE) equals RMSE of the point charge model divided by RMSE when all charges are set to zero. Files for computing RMSE and RRMSE are found in the folder **RRMSE\_and\_RMAE\_scripts**. To compute RMSE and RRMSE in a nonperiodic system, replace the **potential.cube** (Gaussian file) with that for your system, place the atomic charges and atomic dipoles in the **partial\_charges\_data.m** file, and type **calculate\_nonperiodic\_RRMSE** at the MATLAB prompt (or use the example script). To compute RMSE and RRMSE in a periodic system, replace the **LOCPOT** and **POTCAR** files with those for your system, make sure the VASP version is set correctly in the **computational\_parameters.m** file, place the atomic charges and atomic dipoles in the **partial\_charges\_data.m** file, and type **calculate\_periodic\_RRMSE** at the MATLAB prompt (or use the example script). The values of the inner and outer multipliers are found in the **computational\_parameters.m** file and can be changed if desired. The **vdW\_radii.m** file contains the vdW radii, which can also be changed if desired.

### N) Calculating the accuracy of the atomic spin moments for reproducing the magnetic field:

The accuracy of an atomic spin moment (ASM) model for reproducing the magnetic field  $\vec{B}^{spin}(\vec{r})$  can be quantified by the mean absolute error (MAE),

$$MAE = \sum_{\vec{p}} \left| \left( \vec{B}^{spin}(\vec{p}) \right)_{ASM} - \vec{B}^{spin}(\vec{p})_{spindensity} \right| u(\vec{p}) / \sum_{\vec{p}} u(\vec{p})$$

where  $\vec{B}^{spin}(\vec{p})_{ASM}$  and  $\vec{B}^{spin}(\vec{p})_{spindensity}$  are the values at grid point  $\vec{p}$  due to the ASM's and ab initio spin density, respectively. The finite element volume,  $u(\vec{p})$ , for each grid point makes the summations in Eq. (23) equivalent to corresponding integrals over volume.  $\vec{B}^{spin}(\vec{p})_{spindensity}$  is computed by numerically integrating

$$\vec{B}^{spin}(\vec{p}) = \frac{\mu_0}{4\pi} \left( \frac{g_e \mu_B}{2} \right) \oint \left( \frac{3(\vec{p} - \vec{r}')(\vec{m}(\vec{r}') \cdot (\vec{p} - \vec{r}'))}{|\vec{p} - \vec{r}'|^5} - \frac{\vec{m}(\vec{r}')}{|\vec{p} - \vec{r}'|^3} \right) d^3 \vec{r}'$$

over grid points  $\vec{r}'$  inside the surface defined by  $2.4 \times$  vdW radii. The set of grid points,  $\{\vec{p}\}$ , used to compute the MAE are distributed in the volume between surfaces defined by  $3 \times$  and  $4 \times$  vdW radii. The relative mean absolute error (RMAE) is then defined as the MAE for an ASM model divided by the MAE when all ASMs were set to zero.



Files for computing MAE and RMAE are found in the folder **RRMSE\_and\_RMAE\_scripts**. At this time, RMAE is only supported for non-periodic systems. To compute RMAE in a nonperiodic system, replace the **spin\_density.cube** (Gaussian file) with that for your system, place the net atomic spin moments in the **atomic\_spin\_moments\_data.m** file, and type **calculate\_nonperiodic\_RMAE** at the MATLAB prompt (or use the example script). The space between grid points in the **spin\_density.cube** file should be no more than 0.1 Å (0.2 bohr) and enough points around the molecule should be included to encompass the surface defined by 2.4× vdW radii. The total magnetic moment will be computed and printed in the output file; if this value is not within ±0.01 of the exact value you should consider using a finer grid in the **spin\_density.cube** file.

#### O) What to do if you encounter a problem:

- 1) If running using a script, re-run the calculation with Matlab error messages directed to the output file. See details of your shell for how to do this. Some shells allow this to be done by using the redirect **>&** in place of **>** in the script file.
- 2) Check to make sure your calculation has not run out of memory. Computation of charges and spin moments require RAM equal to about ten times the size of your input valence density. For valence density of approximately 200 MB, specify about 2 GM memory. If you think this might be the problem, try increasing the amount of memory in your script file. If you obtain a Matlab error that memory could not be allocated or maximum array size exceeded, then you should consider using a 64 Matlab version that allows arrays of virtually unlimited size.
- 3) Check the grid spacing. The Chargemol program requires adjacent grid points to be at least as close as 0.2 bohr (0.1 Å). The number of valence electrons must also integrate to within 0.1e of the correct values.
- 4) Recheck the values of the parameters in the **chargemol\_job.m** file to make sure the type of calculation (periodic or nonperiodic, etc.) has been correctly specified. *Siesta and CP2K users*: Make sure the number of core electrons has been correctly specified for each element.
- 5) Recheck your input files to make sure they are of the appropriate size. For non-periodic systems, make sure all atoms are at least 4 Å away from the box boundary.
- 6) If you are unable to resolve the problem using 1) to 5), please contact Tom Manz by email: [thomasamanz@gmail.com](mailto:thomasamanz@gmail.com). (Email disguised to prevent spambots from harvesting it.)