Distributed Multipole Analysis of Gaussian wavefunctions
GDMA version 2.3
Anthony J. Stone

1 Introduction

The GDMA program carries out Distributed Multipole Analysis of wavefunctions expressed in terms of Gaussian atomic orbitals. The calculation can be carried out directly in the CamCASP program[1], or by using as input the formatted checkpoint (".fchk") file that can be generated using the Psi4 quantum chemistry code[2] or the Gaussian system of programs[3] . If you find any bugs, please send email to Anthony Stone, ajs1@cam.ac.uk. Comments and corrections would also be welcome. Instructions given here refer to the use of the program with a .fchk file. Most of the information also applies to the CamCASP package; for the differences, see the CamCASP documentation.

The recommended procedure for using the program is to construct a small data file of the following form:

    FILE *checkpointfile* [DENSITY *density-type*]
    [NAMES
        List of atom names, on one or more lines]
    MULTIPOLES
        subcommands, described below
    START

For use with CamCASP, the first line is omitted, and the remaining commands should appear between BEGIN GDMA and END lines in the CamCASP input file. The keywords shown in uppercase may be typed in upper, lower or mixed case. The DENSITY specification is optional; the default is to read the SCF density matrix from the checkpoint file. (Here and later, square brackets denote optional items. The square brackets themselves are not part of the data.) Any other density matrix that appears in the checkpoint file may be specified; if the name contains spaces it must be enclosed in single or double quotes. The NAMES command is optional, but may be needed for some DMA options. If it is provided, the atom names should be listed, separated by spaces, in the order that the atoms appear in the checkpoint file. (At this point the program knows how many atoms there are, but the checkpoint file doesn't include their names.) If the NAMES command is omitted, each atom is given a name which is the chemical symbol corresponding to its atomic number. The MULTIPOLES command may be repeated to obtain DMAs with different options, for example with different choices of multipole sites. The FILE command may be repeated to read another checkpoint file, or a different density from the same file.

Other options that may appear are:

    VERBOSE
This may precede the FILE command, and causes the program to print information about the

data read from the checkpoint file. It is primarily for debugging purposes and should not normally be needed.

ANGSTROM

If needed, this should follow the `FILE` command unless the atom coordinates in the formatted checkpoint file are in ångstrom instead of atomic units. Any length values given in subsequent `MULTIPOLES` sections of the data (see below) will then be taken to be in ångstrom, and coordinates printed by the program will be given in ångstrom also, instead of atomic units.

BOHR

This can be used to cancel the effect of a previous `ANGSTROM` command, but will not normally be needed.

AU

Multipole moments printed by the program will be in atomic units, ea_0^n for moments of rank n. (See sec. 2.1 below for the definition of rank.) This is the default.

SI

Multipole moments printed by the program will be in SI units, Cm^n for moments of rank n, but multiplied by a factor 10^{20+10n}. For example, a dipole moment (rank 1) with a value of 0.7×10^{-30} Cm will be printed as 0.7.

For information on generating the Gaussian formatted checkpoint file, see the Gaussian documentation. In brief, you need a line at the beginning of your Gaussian input of the form

    %chk=*chkfile*

which causes Gaussian to produce a binary checkpoint file called *chkfile*. There is a Gaussian utility `formchk` which converts this into the formatted checkpoint file used by GDMA.

For Psi4, see the Psi4 documentation[2]. In brief, if running under psithon, use something like

```
energy, wfn = psi4.energy("PBE0", return_wfn=True)
psi4.fchk(wfn,"pbe0.fchk")
```

2   Distributed Multipole Analysis

Distributed Multipole Analysis or DMA is a technique for describing a molecular charge distribution by using local multipoles at a number of sites within a molecule. It gives a much more accurate representation of the charge density than a single-point multipole expansion, even for small molecules. For details see §2.1 below. Note that the accuracy of the description depends on the inclusion of site dipoles, quadrupoles and possibly higher moments. If you need a point-charge model, the charges from the distributed multipole analysis are not suitable, but see §3.2 for suggestions. The sites are usually the nuclei, but existing sites can be omitted and new sites added. The multipoles are the charges, dipoles, quadrupoles, etc., up to some limit, which can be as high as rank 10. The limit need not be the same on every atom. For example the following calculation would give multipoles up to rank 4 (hexadecapole) on the oxygen atom, but only up to rank 1 on the hydrogen atoms. (Higher moments on hydrogen atoms are usually very small.) In a case like this, the higher moments on the H atoms are transferred to the nearest atom with a higher limit, here the oxygen, and the overall molecular moments are correct up to the highest

rank specified, rank 4 in this case. (See below, pp. 8ff, for an explanation of the SWITCH and RADIUS commands.)

```
    MULTIPOLES
    LIMIT 4
    LIMIT 1 H
    SWITCH 4.0
    RADIUS H 0.35
    START
```

When used with the output file from a B3LYP calculation on water, using the aug-cc-pVTZ basis set, the output is as shown below. On the oxygen we see the charge (Q00) and the components of the dipole (Q1), the quadrupole (Q2), octopole (Q3) and hexadecapole (Q4) moments, while on the hydrogen we see only the charge and dipole moment, as specified by the LIMIT 1 H data line. The components are expressed in terms of spherical harmonic definitions (see below). Only the non-zero moments are printed.

```
O            x =  0.000000  y =  0.000000  z =  0.116982
             Maximum rank =  4   Relative radius =  0.650
                   Q00   =  -0.276214
|Q1| =   0.484939  Q10   =  -0.484939
|Q2| =   1.669447  Q20   =   0.126597  Q22c =  -1.664640
|Q3| =   3.360888  Q30   =   1.303370  Q32c =   3.097869
|Q4| =   6.929535  Q40   =  -3.612674  Q42c =  -5.030396  Q44c =   3.108402


H            x =  0.000000  y =  0.763556  z = -0.467927
             Maximum rank =  1   Relative radius =  0.350
                   Q00   =   0.138107
|Q1| =   0.031927  Q10   =   0.031774  Q11s =  -0.003124


H            x = -0.000000  y = -0.763556  z = -0.467927
             Maximum rank =  1   Relative radius =  0.350
                   Q00   =   0.138107
|Q1| =   0.031927  Q10   =   0.031774  Q11s =   0.003124


Total multipoles
referred to origin at x =   0.000000,  y =    0.000000,  z =    0.000000
                   Q00   =  -0.000000
|Q1| =   0.726696  Q10   =  -0.726696
|Q2| =   2.164751  Q20   =  -0.276239  Q22c =  -2.147054
|Q3| =   3.481250  Q30   =   1.811733  Q32c =   2.972663
|Q4| =   5.620281  Q40   =  -2.635421  Q42c =  -3.021910  Q44c =   3.938297
```

2.1 Definitions of multipole moments

The spherical harmonic multipole Q_{lk} is defined as

$$Q_{lk} = \int R_{lk}(\mathbf{r})\rho(\mathbf{r})\mathrm{d}^3\mathbf{r}$$

where $\rho(\mathbf{r})$ is the total charge density, and the regular solid harmonics $R_{lk}(\mathbf{r})$ are given in Table 1. For those who prefer the cartesian definitions of multipole moments the relationships between spherical harmonic and cartesian definitions up to hexadecapole are given in Table 2. There are fewer components of the multipole moments in the spherical form than in the cartesian form (except for the charge and dipole) as the components of the cartesian multipoles are not all independent.

The quantities listed as |Qn| in the program output are the magnitudes of the multipole moments, defined by

$$|Q_n| = \left(\sum_k |Q_{nk}|^2\right)^{1/2}.$$

They are independent of axis system, and $|Q_1|$ coincides with the usual expression for the magnitude of the dipole moment.

As well as the distributed multipoles, the program prints the total multipoles referred to the origin of the coordinate frame. Components which are zero are not printed, except for the total charge.

All moments are usually given in atomic units, ea_0^k for moments of rank k, but it is possible to have them printed in SI units instead (SI command, p. 2).

2.2 Multipole allocation algorithms

The original distributed multipole analysis, as set out in the bibliography[4–6], starts from the expansion of the electron density in terms of the primitive gaussian basis functions $\chi_i(\boldsymbol{r})$:

$$\chi_i(\boldsymbol{r}) = N_i x_i^{a_i} y_i^{b_i} z_i^{c_i} \exp[-\zeta_i(\boldsymbol{r}_i)^2], \tag{1}$$

where $\boldsymbol{r}_i = \boldsymbol{r} - \boldsymbol{A}_i$ is the electron position relative to the position $\boldsymbol{A}_i$ of the primitive Gaussian, ζ_i is the exponent and N_i is a normalising factor. The electron density is then

$$\rho(\boldsymbol{r}) = \sum_{ij} D_{ij}\chi_i(\boldsymbol{r})\chi_j(\boldsymbol{r}). \tag{2}$$

Boys[7] showed that a product of gaussians $\chi_i(\boldsymbol{r})\chi_j(\boldsymbol{r})$ can be expressed as a gaussian function centred at $\boldsymbol{P}_{ij} = (\zeta_i\boldsymbol{A}_i + \zeta_j\boldsymbol{A}_j)/(\zeta_i + \zeta_j)$. The multipole moments of the associated charge distribution can be calculated exactly, and in standard DMA they are represented by a multipole expansion about the nearest site. This procedure is very fast and efficient, and gives an excellent representation of the molecular charge distribution.

If the basis set includes very diffuse functions, however, overlap densities involving them extend over many atoms. In carbon dioxide, for instance, the product of a diffuse s function on

4

Table 1: Regular solid harmonics

$$R_{00} = 1$$

$$R_{10} = z$$

$$R_{11c} = x$$

$$R_{11s} = y$$

$$R_{20} = \tfrac{1}{2}(3z^2 - r^2)$$

$$R_{21c} = \sqrt{3}xz$$

$$R_{21s} = \sqrt{3}yz$$

$$R_{22c} = \sqrt{\tfrac{3}{4}}(x^2 - y^2)$$

$$R_{22s} = \sqrt{3}xy$$

$$R_{30} = \tfrac{1}{2}(5z^3 - 3zr^2)$$

$$R_{31c} = \sqrt{\tfrac{3}{8}}x(5z^2 - r^2)$$

$$R_{31s} = \sqrt{\tfrac{3}{8}}y(5z^2 - r^2)$$

$$R_{32c} = \sqrt{\tfrac{15}{4}}z(x^2 - y^2)$$

$$R_{32s} = \sqrt{15}xyz$$

$$R_{33c} = \sqrt{\tfrac{5}{8}}(x^3 - 3xy^2)$$

$$R_{33s} = \sqrt{\tfrac{5}{8}}(3x^2y - y^3)$$

$$R_{40} = \tfrac{1}{8}(35z^4 - 30z^2r^2 + 3r^4)$$

$$R_{41c} = \sqrt{\tfrac{5}{8}}(7xz^3 - 3xzr^2)$$

$$R_{41s} = \sqrt{\tfrac{5}{8}}(7yz^3 - 3yzr^2)$$

$$R_{42c} = \sqrt{\tfrac{5}{16}}(x^2 - y^2)(7z^2 - r^2)$$

$$R_{42s} = \sqrt{\tfrac{5}{4}}xy(7z^2 - r^2)$$

$$R_{43c} = \sqrt{\tfrac{35}{8}}z(x^3 - 3xy^2)$$

$$R_{43s} = \sqrt{\tfrac{35}{8}}z(3x^2y - y^3)$$

$$R_{44c} = \sqrt{\tfrac{35}{64}}(x^4 - 6x^2y^2 + y^4)$$

$$R_{44s} = \sqrt{\tfrac{35}{4}}(x^3y - xy^3)$$

one O atom and a diffuse p_z function with slightly larger exponent on the C atom would be formally centred between the C and O, but slightly nearer to the C atom, and would give rise to charge and dipole contributions which would be allocated to the C atom. However, much of the electron density would in fact be in the region of the O atom, and it would extend to some extent over all three atoms. It would be more satisfactory to apportion the charge density between the atoms in some way.

Methods for doing this have been developed in recent years for handling the integrals of density functional theory. Version 2 of the GDMA program[8] uses these methods to calculate the multipole contributions arising from the overlap of diffuse primitive functions. Such an approach is not needed for compact basis functions (those with large ζ) because the overlap densities involving such functions are well localised in space, and their allocation to the nearest multipole site is entirely satisfactory. Moreover the grid-based quadrature used for the diffuse functions is less satisfactory for highly-peaked functions.

Consequently the program can now use both methods. If the sum of exponents $\zeta_i + \zeta_j$ for a pair of primitive functions χ_i and χ_j is less than a switch value Z, the grid-based analysis is used, and otherwise the original DMA method is used. If $Z = 0.0$, the original DMA algorithm is used throughout. However the default value is $Z = 4.0$, and this is recommended for normal use, especially with basis sets containing very diffuse functions. The parameters for the grid-based quadrature can also be specified, but the default values should be satisfactory in most cases.

Table 2: Relationship between spherical-tensor and cartesian-tensor form of multipole moments

$$\mu_z = Q_{10}$$
$$\mu_x = Q_{11c}$$
$$\mu_y = Q_{11s}$$

$$\Theta_{xx} = -\tfrac{1}{2}Q_{20} + \tfrac{1}{2}\sqrt{3}Q_{22c}$$
$$\Theta_{yy} = -\tfrac{1}{2}Q_{20} - \tfrac{1}{2}\sqrt{3}Q_{22c}$$
$$\Theta_{zz} = Q_{20}$$
$$\Theta_{xy} = \tfrac{1}{2}\sqrt{3}Q_{22s}$$
$$\Theta_{xz} = \tfrac{1}{2}\sqrt{3}Q_{21c}$$
$$\Theta_{yz} = \tfrac{1}{2}\sqrt{3}Q_{21s}$$

$$\Omega_{xxx} = \sqrt{\tfrac{5}{8}}Q_{33c} - \sqrt{\tfrac{3}{8}}Q_{31c}$$
$$\Omega_{xxy} = \sqrt{\tfrac{5}{8}}Q_{33s} - \sqrt{\tfrac{1}{24}}Q_{31s}$$
$$\Omega_{xyy} = -\sqrt{\tfrac{5}{8}}Q_{33c} - \sqrt{\tfrac{1}{24}}Q_{31c}$$
$$\Omega_{yyy} = -\sqrt{\tfrac{5}{8}}Q_{33s} - \sqrt{\tfrac{3}{8}}Q_{31s}$$
$$\Omega_{xxz} = \sqrt{\tfrac{5}{12}}Q_{32c} - \tfrac{1}{2}Q_{30}$$
$$\Omega_{xyz} = \sqrt{\tfrac{5}{12}}Q_{32s}$$
$$\Omega_{yyz} = -\sqrt{\tfrac{5}{12}}Q_{32c} - \tfrac{1}{2}Q_{30}$$
$$\Omega_{xzz} = \sqrt{\tfrac{2}{3}}Q_{31c}$$
$$\Omega_{yzz} = \sqrt{\tfrac{2}{3}}Q_{31s}$$
$$\Omega_{zzz} = Q_{30}$$

$$\Phi_{xxxx} = \tfrac{3}{8}Q_{40} - \tfrac{1}{4}\sqrt{5}Q_{42c} + \tfrac{1}{8}\sqrt{35}Q_{44c}$$
$$\Phi_{xxxy} = \tfrac{1}{8}(-\sqrt{5}Q_{42s} + \sqrt{35}Q_{44s})$$
$$\Phi_{xxyy} = \tfrac{1}{8}Q_{40} - \tfrac{1}{8}\sqrt{35}Q_{44c}$$
$$\Phi_{xyyy} = -\tfrac{1}{8}(\sqrt{5}Q_{42s} + \sqrt{35}Q_{44s})$$
$$\Phi_{yyyy} = \tfrac{3}{8}Q_{40} + \tfrac{1}{4}\sqrt{5}Q_{42c} + \tfrac{1}{8}\sqrt{35}Q_{44c}$$
$$\Phi_{xxxz} = \tfrac{1}{16}(-3\sqrt{10}Q_{41c} + \sqrt{70}Q_{43c})$$
$$\Phi_{xxyz} = -\tfrac{1}{16}(-\sqrt{10}Q_{41s} + \sqrt{70}Q_{43s})$$
$$\Phi_{xyyz} = -\tfrac{1}{16}(\sqrt{10}Q_{41c} + \sqrt{70}Q_{43c})$$
$$\Phi_{yyyz} = -\tfrac{1}{16}(3\sqrt{10}Q_{41s} + \sqrt{70}Q_{43s})$$
$$\Phi_{xxzz} = -\tfrac{1}{2}Q_{40} + \tfrac{1}{4}\sqrt{5}Q_{42c}$$
$$\Phi_{xyzz} = \tfrac{1}{4}\sqrt{5}Q_{42s}$$
$$\Phi_{yyzz} = -\tfrac{1}{2}Q_{40} - \tfrac{1}{4}\sqrt{5}Q_{42c}$$
$$\Phi_{xzzz} = \sqrt{\tfrac{5}{8}}Q_{41c}$$
$$\Phi_{yzzz} = \sqrt{\tfrac{5}{8}}Q_{41s}$$
$$\Phi_{zzzz} = Q_{40}$$

2.3 Options for multipole analysis

The `MULTIPOLES` command has various subcommands, two of which, `LIMIT` and `RADIUS`, were included in the example dataset. The `LIMIT` command sets the maximum rank for the DMA analysis. For example, `LIMIT 4` instructs the program to calculate multipoles up to hexadecapole only. The highest rank which can be calculated for non-linear molecules is 10, i.e. `LIMIT 10`. `LIMIT 1` would calculate only charge and dipole on each site, `LIMIT 2` would include the quadrupole moments, and so on. It is also possible to set different limits for different sites. For example

```
MULTIPOLES
LIMIT 4
LIMIT 1 H
START
```

would calculate multipoles up to hexadecapole on all sites except those called 'H', where the expansion would terminate with the dipole moments. If this dataset is used for water, as in the example, the effects of the higher moments on the H atoms are transferred to the O atom, so that the overall multipole moments for the whole molecule are correct up to the highest rank specified. This is usually a sensible procedure for H atoms, where the higher moments are small. The limit for named atoms is applied to all atoms with the name given, so if different limits are wanted for atoms of the same element, the `NAMES` command must be used to give them distinctive names. The general form of the `LIMIT` command is '`LIMIT` *n name name* ...' or '`LIMIT` *n* `ALL` *aa*'. The latter form applies the limit to all sites with names starting with '*aa*'. Site names are always case-sensitive.

The list of sites can be modified using `ADD` and `DELETE`. Extra sites can be included in the analysis by using the `ADD` command:

   `ADD` *name x y z* [`LIMIT` *n*] [`RADIUS` *r*]

where *x*, *y* and *z* are the cartesian coordinates of the new site (in atomic units, unless the `ANGSTROM` command has been used). Items in square brackets are optional. The optional `LIMIT` *n* sets the maximum rank for this site; if it is omitted, the current maximum rank is used. The `RADIUS` option, if present, sets the radius (see below). If it is omitted, a value of 0.65 Å is used.

`DELETE` removes all sites with the name specified. For example,

```
MULTIPOLES
DELETE H
START
```

would remove from the analysis all sites called 'H'. This would sometimes be done in large molecules to avoid an over-proliferation of expansion centres. For example one might want to avoid using the methyl hydrogen atoms as multipole sites in a large molecule. To delete the methyl hydrogens in methanol but retain the OH hydrogen (a common situation where hydrogen bonding is involved) it would be necessary to use different names for the methyl and OH hydrogens, perhaps 'H' and 'Ho' respectively, and then `DELETE H` would have the required effect.

If you use `DELETE ALL` then all sites are removed. In this case you will need to use `ADD` to provide at least one site for the expansion. This provides a way to get a single-site expansion referred to a site other than the origin of coordinates. However a simpler way is to use the

command
        ORIGIN o_x o_y o_z
to specify the origin for the overall multipole moments.

It should perhaps be emphasized once again that whatever sites are added or deleted, the overall multipole moments will be correct up to the maximum rank of any site that remains. However, if too few sites are used, the resulting multipole expansion of the electrostatic potential or of the electrostatic interaction between molecules may be inaccurate. See my book for a fuller discussion[6].

The switch between the original DMA algorithm and the new grid-based quadrature can be controlled by the command
        SWITCH Z
The default value is 4.0. To use the original algorithm, use SWITCH 0.0, but it is important in this case to check that the results are sensible.

The GRID command controls parameters for the quadrature grid:
        GRID *options*
The options are
        LEBEDEV n
Use Lebedev angular quadrature[9] with at least n points for each atom. Lebedev quadrature is the default, but this command is needed if the number of points is to be changed from the default of 590. Note that a high value is needed, especially if high-rank multipoles are required, because they are quite sensitive to the angular behaviour of the charge density.
        GAUSS-LEGENDRE n
Use Gauss–Legendre quadrature in θ and equally-spaced intervals in ϕ, with the total number of points chosen to be at least n.
        RADIAL n
Use n points for the radial quadrature (default 80). This value needs to be large, especially if high-rank multipoles are required, because they are strongly influenced by the electron density at large distances from the nuclei.
        SMOOTHING m
This specifies the integer parameter used in the Becke procedure[10] for calculating the weights of points assigned to each atom. The default value is 3; this is smaller than is usually recommended for DFT calculations, because a softer transition between atoms is more suitable for multipole moment calculations.

It is usual in setting up the grid for grid-based quadrature to assign different radii to different elements. These radii are used in scaling the radial quadrature grid[11]. The radii used in standard density-functional calculations are usually Bragg–Slater covalent radii[12], though the value used for hydrogen is twice the Bragg–Slater value. These are used here in the usual way for scaling the radial grid, but they are not suitable for partitioning the density between atoms in the distributed multipole analysis — they lead to large and implausible atom charges — and for this purpose the GDMA program sets all the atom radii to be equal by default, except for hydrogen (see below). The use of equal radii for all sites is the most efficient choice from the point of view of convergence of the resulting multipole expansion of the electrostatic potential[6]. The radius used is 0.65 Å. Additional sites specified using the ADD command are

also assigned a radius of 0.65 Å unless some other value is explicitly specified. Different values may be assigned to existing sites using the option

    `RADIUS` n_1 r_1 n_2 r_2 ...

which assigns the radius r_1 to every site with the name n_1, radius r_2 to sites called n_2, and so on. The values chosen for the radii are not critical — any reasonable choice will give multipoles that describe the electrostatic potential accurately — but some experimentation may be needed to obtain atom charges that correspond to chemical intuition. In particular, a radius of 0.325 Å for hydrogen has been found to give more acceptable values than the default of 0.65 Å, especially where hydrogen-bonding is involved, **and this is now the default for hydrogen**. That is, any atom site with nuclear charge 1 is initially assigned a radius of 0.325 Å.

The atom radii control the allocation of multipoles as follows. For any multipole that is to be moved to one of the specified multipole sites, the distance to each site is evaluated and divided by the radius for that site. The multipole is then moved to the site with the smallest resulting scaled value.

Generally speaking, it is best to move multipoles as short a distance as possible, so the multipoles as initially calculated, either by the Gauss-Hermite quadrature or as contributions to a grid-based treatment, are generally best moved to the nearest multipole site. This is achieved if the atom radii are all the same. However, in the case of hydrogen, results that are more in line with chemical intuition are obtained with the smaller radius. If in other cases the results don't seem to agree with chemical intuition, some adjustment of other radii may give more intuitive results. For example, a radius of 1.11 Å has been found suitable for Cl in methyl chloride.

The command

    `PUNCH [APPEND] [RANK p]` [*punch-file*]

causes the program to produce a summary of output in the specified file in a form suitable for reading in to the Orient program[13]. This output includes only multipoles up to rank p (default 5), or less for any site for which a lower limit has been specified. The numerical values will be in atomic units, even if printed values are in SI. If the file name is omitted, and a punch file has been specified for an immediately preceding `MULTIPOLES` command in the same job, the punch output is appended to the same file.

`APPEND` is optional. If it is omitted, any existing file with the specified name is overwritten, unless it has already been opened for punch output from an immediately preceding `MULTIPOLES` command in the same job, in which case `APPEND` is assumed. If `APPEND` is specified or assumed, and the file exists, the new output is appended to it.

The `MULTIPOLES` command may be used any number of times to construct different multipole representations of the electron density. If the checkpoint file contains more than one density matrix — perhaps an SCF density and an MP2 density — the `FILE` command can be repeated, specifying a different density, to re-read the checkpoint file. Subsequent `MULTIPOLES` commands will then use that density.
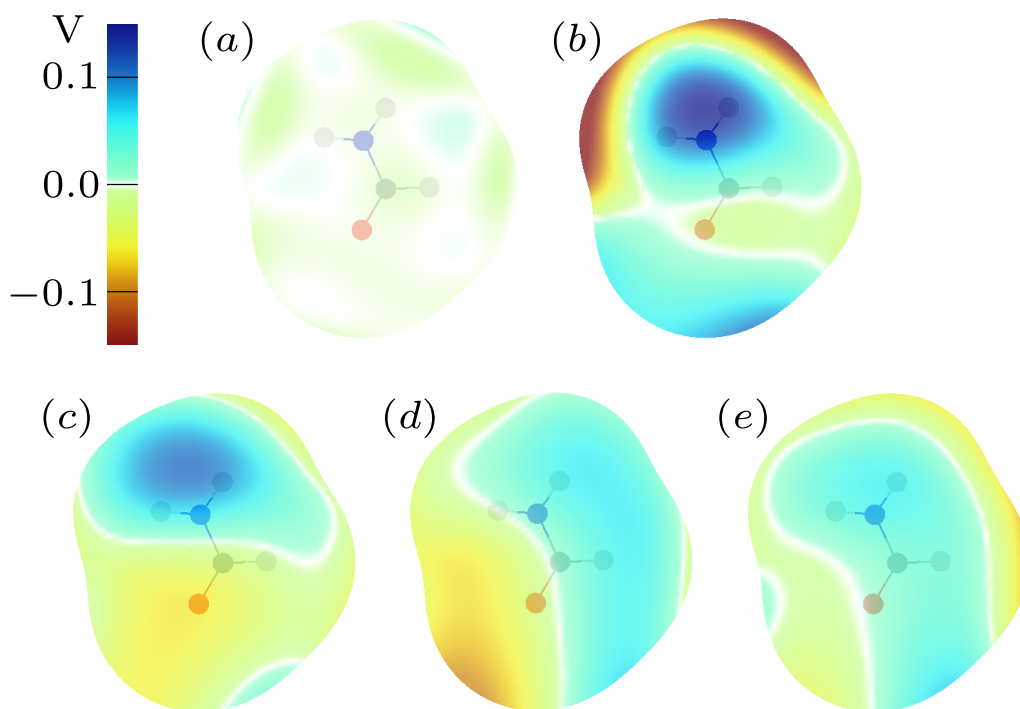
Figure 1: Errors in potentials due to point-charge and distributed-multipole models for the electrostatic potential of formamide on a surface at 1.8 van der Waals radii. The maps show the difference between the exact potentials calculated from an aug-cc-pVQZ wavefunction and (a) rank 5 DMA (b) DMA charges only, (c) charges fitted to DMA rank 5 using MULFIT, (d) charge of iterated stockholder atoms (ISA), (e) CHELPG potential-derived charges. For an interacting charge of $0.5e$, 0.1 V corresponds to about $5\,\mathrm{kJ\,mol^{-1}}$.

3   Important notes

3.1   Algorithms

The algorithm originally used by the program is both exact and very fast, because it uses an exact and very efficient Gauss-Hermite quadrature. The new version normally uses a grid-based quadrature for integrations involving diffuse functions, and is very much slower, because it is necessary to use a fine grid, and even with a fine grid it is not exact. An estimate of the errors may be obtained by carrying out a calculation for a symmetrical molecule where some multipoles should be zero by symmetry. They will normally be much smaller than any uncertainties arising from approximations in the original ab initio calculation.

The original integration algorithm can be used by setting "SWITCH 0", as explained above. The calculation then runs at full speed. The multipoles may not correspond so well with chemical intuition, however, and with large basis sets incorporating diffuse primitive functions they may be quite unreasonable. The difference in electrostatic potential between models obtained with different swiitch values may be explored using my Orient program.

3.2 Point-charge models

If you need a point-charge model, do not use the atom charges from GDMA. The effects of atomic distortions caused by chemical bonding are described in GDMA by the dipoles and higher moments, and simply omitting those will give very poor results. For good results you need to include dipoles and quadrupoles at least. However, some applications can only handle point charge models, so it is sometimes necessary to provide them. There are several possibilities, illustrated in Fig. 1, which shows the difference between the electrostatic potential of each of several models and the exact potential evaluated as the interaction energy of a unit charge with the electron density. The electron density and all models were evaluated using the aug-cc-pVQZ basis, and the potential maps were prepared using the ORIENT program[14]. Fig. 1(a) shows the difference for the DMA to rank 5, for which the RMS error is 0.0026V. Fig. 1(b) shows the potential due to the point charges of the DMA model only. This is clearly unsatisfactory, with an r.m.s error of 0.07V and a maximum error of nearly 0.2V. One recommended procedure is to use the MULFIT program of Winn *et al.* [15, 16], which approximates the atomic dipoles and higher moments by means of charges on neighbouring atoms. (MULFIT can also be used to refine a more detailed description, for example with multipoles up to quadrupole, by approximating the contributions of the octopoles and hexadecapoles.) Fig. 1(c) illustrates the improvement that can be obtained in this way; the r.m.s. error is 0.041V. Fig. 1(d) was obtained using the charges from an iterated stockholder atom (ISA) description[17], calculated using the CAMCASP program of Misquitta & Stone[1], and Fig. 1(e) was obtained using CHELPG charges, calculated using Gaussian03. The r.m.s. differences for these cases are 0.044V and 0.034V respectively. Note that the ISA method yields a complete distributed-multipole description, not just the charge model used here.

The MULFIT package is included with the GDMA distribution, by permission of its authors. See the instructions included with that package for details of its use.

4 Example data files

A number of examples are provided in the `examples` directory and its subdirectories, together with output files. The script `run.tests` in the `examples` directory runs these calculations automatically and compares the results with the output files provided.

5 Program limits

In version 2.2.02 and later, the arrays used by the program are allocated as required, so there should be no problems with large molecules or large basis sets. The maximum number of sites is arbitrarily set at 16 more than the number of atoms, and this should be sufficient, but if more sites are needed it will be necessary to increase the value of `nextra` near the top of the gdma.f90 file and recompile.

6 Citation

Please use the citation as in Ref. 8 when referring to the program.

7 Revision notes

Version 2.3 allows for basis sets that include *h* functions. Both Gaussian and Psi4 can produce the necessary fchk files. An unnecessary limit of 16 on contraction depth has been removed.

Version 2.2.02 uses dynamic allocation of arrays, so that there are no arbitrary limits on the size of the molecule or the basis set. There is an arbitrary limit on the number of sites, as noted above, but it should be adequate for most if not all cases.

Version 2.2 incorporates the minor but significant change that the radius assigned by default to hydrogen atoms is half the default radius for other atoms. Calculations involving H atoms with default radii will give results that are different from earlier versions.

Version 2.1 provides for printing multipole moments in SI units as an alternative to the standard atomic units.

Version 2.0 This version of the program can handle basis functions up to *g*. It also provides for real-space apportionment of the charge density arising from low-exponent primitives, instead of the allocation of all multipoles from a particular primitive-function overlap density to the nearest site.

Version 1.3 Explicit type declarations introduced. This dealt with an apparent bug in the Portland compiler, which gave wrong results when implicit declarations were used.

Version 1.2 Modified to handle new features in the Gaussian03 formatted fchk files.

References

[1] A. J. Misquitta and A. J. Stone, CamCASP: a program for studying intermolecular interactions and for the calculation of molecular properties in distributed form, University of Cambridge, (2023), https://github.com/ajmisquitta/camcasp-bin.

[2] D. G. A. Smith, L. A. Burns, A. C. Simmonett *et al.*, J. Chem. Phys. 152 (2020) 184108.

[3] M. J. Frisch, G. W. Trucks, H. B. Schlegel, G. E. Scuseria, M. A. Robb, J. R. Cheeseman, G. Scalmani, V. Barone, G. A. Petersson, H. Nakatsuji, X. Li, M. Caricato, A. V. Marenich, J. Bloino, B. G. Janesko, R. Gomperts, B. Mennucci, H. P. Hratchian, J. V. Ortiz, A. F. Izmaylov, J. L. Sonnenberg, D. Williams-Young, F. Ding, F. Lipparini, F. Egidi, J. Goings, B. Peng, A. Petrone, T. Henderson, D. Ranasinghe, V. G. Zakrzewski, J. Gao, N. Rega, G. Zheng, W. Liang, M. Hada, M. Ehara, K. Toyota, R. Fukuda, J. Hasegawa, M. Ishida, T. Nakajima, Y. Honda, O. Kitao, H. Nakai, T. Vreven, K. Throssell, J. A. Montgomery, Jr., J. E. Peralta, F. Ogliaro, M. J. Bearpark, J. J. Heyd, E. N. Brothers, K. N. Kudin, V. N. Staroverov, T. A. Keith, R. Kobayashi, J. Normand, K. Raghavachari, A. P. Rendell, J. C. Burant, S. S. Iyengar, J. Tomasi, M. Cossi, J. M. Millam, M. Klene, C. Adamo, R. Cammi, J. W. Ochterski, R. L. Martin, K. Morokuma, O. Farkas, J. B. Foresman and D. J. Fox (2016), Gaussian Inc. Wallingford CT.

[4] A. J. Stone, Chem. Phys. Lett. 83 (1981) 233.

[5] A. J. Stone and M. Alderton, Molec. Phys. 56 (1985) 1047.

[6] A. J. Stone, The Theory of Intermolecular Forces, (Oxford University Press, Oxford, 2013), 2nd edn.

[7] S. F. Boys, Proc. Roy. Soc. A 200 (1950) 542.

[8] A. J. Stone, J. Chem. Theory Comput. 1 (2005) 1128.

[9] V. I. Lebedev and D. N. Laikov, Doklady Mathematics 59 (1999) 477.

[10] A. D. Becke, J. Chem. Phys. 88 (1988) 2547.

[11] C. W. Murray, N. C. Handy and G. J. Laming, Molec. Phys. 78 (1993) 997.

[12] J. C. Slater, J. Chem. Phys. 41 (1964) 3199.

[13] A. J. Stone, A. Dullweber, O. Engkvist, E. Fraschini, M. P. Hodges, A. W. Meredith, D. R. Nutt, P. L. A. Popelier and D. J. Wales, ORIENT: a program for studying interactions between molecules, version 5.0, (2018), https://gitlab.com/anthonyjs/orient.git.

[14] A. J. Stone, A. Dullweber, O. Engkvist, E. Fraschini, M. P. Hodges, A. W. Meredith, D. R. Nutt, P. L. A. Popelier and D. J. Wales, ORIENT: a program for studying interactions between molecules, version 4.8, University of Cambridge, (2012), http://www-stone.ch.cam.ac.uk/programs.html#Orient.

[15] P. J. Winn, G. G. Ferenczy and C. A. Reynolds, J. Phys. Chem. A 101 (1997) 5437.

[16] G. G. Ferenczy, P. J. Winn and C. A. Reynolds, J. Phys. Chem. A 101 (1997) 5446.

[17] A. J. Misquitta, A. J. Stone and F. Fazeli, J. Chem. Theory Comput. 10 (2014) 5405.