

## Chapter 6

# XDPROP – One-Electron Properties based on the Multipole Representation of the ED

## 6.1 Overview

XDPROP is a program for analysing a STATIC ED which has been obtained in the form of a nuclear-centered multipole expansion. It is based on the earlier programs LSPROP (by Sean Howard & Paul Mallinson) and MOLPROP (by Zhengwei Su). Many of the features in the static properties package are not new. Existing programs such as VALRAY [1] (by Mark Spackman and Bob Stewart), POP (by Brian Craven), LSEXP (by Fred Hirshfeld), and MOLLY[2] by Niels Hansen and Philip Coppens) have the ability to compute electrostatic potentials, dipole moments and carry out critical point analyses of the total density. For accounts of the state-of-the-art of charge density applications, the reader is referred to recent books and reviews [3].

The version of XDPROP in XD2016 is substantially upgraded from the previous release (4.10), and is under continuous development. New features include the calculation of lattice energies and interaction energies (**LATEN** & **INTEREN**), gradient trajectory plots of  $\rho$  and the esp (**TRAJPLT**), vector plots of the electric field (**VECPLOT**), atom basin integration of properties (**TOPINT**) and calculation of the local and integrated source function (**SOURCE**). Experienced users of XD are advised to read this Chapter carefully, as a number of options have been significantly modified.

### 6.1.1 Abbreviations

This chapter uses the following notation and abbreviations:  $\rho$  (total electron density); CP (Critical Point);  $\Delta\rho$  (Deformation Density); IAM (Independent Atom Model);  $V(\mathbf{r})$  (electrostatic potential);  $v(r)$  (nuclear potential); BP (Bond Path - a line of maximum charge density connecting two nuclei); PD charges (electrostatic Potential-Derived charges);  $E$  (Electric field); EFG (Electric Field Gradient); ST (Slater-Type, as of a radial function); HF (Hartree-Fock); LSQ (Least Squares, as of a refinement procedure); ESP (ElectroStatic Potential).

### 6.1.2 Units

XDPROP output is in Å, electrons and degrees. Exceptions include the  $\zeta_i$ -values (exponents of Slater-type radial functions), which are conventionally reported in Bohr<sup>-1</sup>, and multipole moments (Debye and Debye-Å).

### 6.1.3 Files used and created by XDPROP

|                  |  |
|------------------|--|
| Input:           | <code>xd.mas</code> , <code>xd.res</code> , <code>xd.bnk_*</code>  |
| Optional input:  | <code>xd.inp</code> (if <code>xd.res</code> does not exist)<br><code>xd.cov</code> (if errors are to be calculated)  |
| Output:          | <code>xd_pro.out</code>  |
| Optional output: | <code>xd_&lt;prop&gt;.cps</code> critical points with eigenvectors of Hessian<br><code>xd_&lt;prop&gt;.grd</code> 2D or 3D XD grid files from MAP/CUBE<br><code>xd_&lt;prop&gt;.cube</code> Gaussian-style grid files from MAP/CUBE<br><code>xd.pth</code> bond path file for visualisation with XDGRAPH<br><code>&lt;prop&gt;_traj.ps</code> PostScript graphic file from TRAJPLT<br><code>&lt;prop&gt;_traj.cgm</code> CGM graphic file from TRAJPLT<br><code>xd_&lt;prop&gt;.dat</code> vector map for visualisation with PLOTMTV |

```

xd_bubble.spf critical points in Laplacian for PLATON
min16.inp atomic moments in spherical tensor notation for MIN16
orient.inp atomic moments in spherical tensor notation for ORIENT

```

### 6.1.4 Coordinate systems

Although XDLSM uses local coordinate systems on the pseudoatoms, XDPROP works almost exclusively in a single, "global" system of Cartesian coordinates  $\mathbf{r}$  generated from the fractional co-ordinates  $\mathbf{r}_f$  in `xd.res` by the transformation

$$\mathbf{r} = \mathbf{M} \mathbf{r}_f \quad (\text{Eq. 6.1})$$

where  $\mathbf{M}$  is the matrix [4]

$$\begin{bmatrix} b \cos \gamma & c \cos \beta & \\ 0 & b \sin \gamma & c(\cos \alpha - \cos \beta \cos \gamma) / \sin \gamma \\ 0 & 0 & \frac{c}{\sin \gamma} \sqrt{\sin^2 \gamma - (\cos^2 \alpha + \cos^2 \beta + \cos^2 \gamma - 2 \cos \alpha \cos \beta \cos \gamma)} \end{bmatrix}$$

This is particularly important in considering dipole and quadrupole moments computed with XDPROP, which are computed in this frame.

### 6.1.5 Current Limitations

- The calculation of errors on  $\rho$  and  $-\nabla^2\rho$  has some severe limitations at the moment, namely: (i) only contributions from the multipole populations are currently taken into account (*i.e.* not  $\kappa$ 's or coordinates) (ii) contributions to the error from symmetry-generated atoms are not taken into account.
- The potential-derived charge fitting subroutine **QFIT** assumes that the fragment is molecular (neutral), so it will give nonsensical results for ions.
- It is only possible to compute errors on the dipole moment,  $\rho$  or  $-\nabla^2\rho$  if Slater-type radial functions have been used for all multipoles.

## 6.2 General Instructions for XDPROP

The Master file entry for XDPROP should begin with `MODULE *XDPROP` and be terminated by the `END XDPROP` line. In general, the order of directives is not important, with a few exceptions. The ordering given here follows that written into the default `xd.mas` by XDINI. These general instructions apply to all sub options in XDPROP

### 6.2.1 SIZE

#### **SIZE** *napl* *napl*

In XD2006, many of the large arrays in XDPROP are dynamically allocated, according to the number of atoms actually present in the asymmetric unit. However, it is algorithmically inconvenient to dynamically allocate the arrays which use the parameter *napl* (the maximum number of symmetry equivalent positions *for any single atom*, generated by the `APPLY/GROUP/OMIT` instructions). A relatively modest value (*napl*=8) is used for these arrays by default. The **SIZE** directive may be used to increase this default value, if it is too small for the problem under investigation. *If used, the SIZE directive must be the first one in the XDPROP section.*

### 6.2.2 MODEL

#### MODEL (\*)iam (\*)multipole

Only two options are currently available - to use the **multipole** model (or some modification of it) which was applied in the refinement; or to generate the **iam** (independent atom model) in its place. In the latter case, multipole populations ( $l \geq 1$ ) are set to zero, monopole populations are set to the free atom values, and the  $\kappa$  parameters are set to unity.

### 6.2.3 PROPERTY

#### PROPERTY \*prop

XDPROP will analyse the property which is flagged with an asterisk (\*) on the **PROPERTY** card. If no property is flagged, then the default property (rho) is chosen. The following properties are currently available:

|                |   |                           |
|----------------|---|---------------------------|
| <b>rho</b>     | Total density ( $\rho$ )                                | $\text{e}\text{\AA}^{-3}$ |
| <b>gradrho</b> | Magnitude of gradient vector of $\rho$ ( $\nabla\rho$ ) | $\text{e}\text{\AA}^{-4}$ |
| <b>d2rho</b>   | Laplacian of $\rho$ ( $\nabla^2\rho$ )                  | $\text{e}\text{\AA}^{-5}$ |
| <b>nucpot</b>  | Nuclear potential ( $V(\mathbf{r})$ )                   | $\text{e}\text{\AA}^{-1}$ |
| <b>core</b>    | Core density  | $\text{e}\text{\AA}^{-3}$ |
| <b>valence</b> | Valence density   | $\text{e}\text{\AA}^{-3}$ |
| <b>defden</b>  | Deformation density ( $\Delta\rho$ )                    | $\text{e}\text{\AA}^{-3}$ |
| <b>sigrho</b>  | Error in $\rho$   | $\text{e}\text{\AA}^{-3}$ |
| <b>siglap</b>  | Error in $\nabla^2\rho$                                 | $\text{e}\text{\AA}^{-5}$ |
| <b>esp</b>     | Electrostatic potential $V(\mathbf{r})$                 | $\text{e}\text{\AA}^{-1}$ |
| <b>ef</b>      | Electric field vector                                   | $\text{e}\text{\AA}^{-2}$ |
| <b>efg</b>     | Electric field gradient tensor                          | $\text{e}\text{\AA}^{-3}$ |
| <b>oep</b>     | One electron potential                                  | $\text{\AA}^{-2}$         |
| <b>nci</b>     | Reduced density gradient $s(\mathbf{r})$                | unitless                  |
| <b>dori</b>    | Density overlap regions indicator                       | unitless                  |
| <b>sedd</b>    | Single exponential decay detector                       | unitless                  |
| <b>srho</b>    | Signed rho = $\rho \times \text{sign}(\lambda_2)$       | $\text{e}\text{\AA}^{-3}$ |
| <b>eig2</b>    | $\lambda_2$ of electron density Hessian                 | $\text{e}\text{\AA}^{-5}$ |

The following properties are calculated using the Kirzhnits approximation for the kinetic energy density. **NOTE** *This is a very poor approximation in regions of high electron density!*

|            |                                    |                           |
|------------|------------------------------------|---------------------------|
| <b>elf</b> | Electron Localisation Function     | unitless                  |
| <b>lol</b> | Localized Orbital Locator          | unitless                  |
| <b>ked</b> | Kinetic Energy Density (Kirzhnitz) | $\text{h}\text{\AA}^{-3}$ |
| <b>ped</b> | Potential Energy Density           | $\text{h}\text{\AA}^{-3}$ |
| <b>ted</b> | Total Energy Density               | $\text{h}\text{\AA}^{-3}$ |

The core/valence decomposition of the electron density is based on the orbital occupations given in `xd.bnk_*` or in the **SCAT** table in the master file (see Section 4.6.2)

$\Delta\rho$  is the standard deformation density, i.e. the difference  $\rho - \rho_{IAM}$ .

The electrostatic potential  $V(\mathbf{r})$  is defined as

$$V(\mathbf{r}) = \sum_j \frac{Z_j}{|\mathbf{r} - \mathbf{R}_j|} - \int \frac{\rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r}' \quad (\text{Eq. 6-2})$$

where  $\mathbf{R}_j$  and  $Z_j$  are the position and charge of the  $j$ -th nucleus, respectively. Outside the Van der Waals surface, it may be computed from an expansion in the multipole moments of individual pseudoatoms [33,34]

$$V(\mathbf{r}) = \sum_j \frac{Z_j}{|\mathbf{r} - \mathbf{R}_j|} + \sum_j \left[ \frac{q_j}{r_j} + \frac{\mathbf{p}_j \cdot \mathbf{e}_j}{|\mathbf{r}_j|^3} + \frac{1}{2} \sum_k \sum_l Q_{jkl} \frac{r_{ji} r_{jk}}{|\mathbf{r}_j|^5} + \dots \right] \quad (\text{Eq. 6-3})$$

where  $q, \mathbf{p}, Q, \dots$  represent pseudoatom moments. A number of experimental charge density analyses using this type of multipolar representation of the potential have been reported [35-38]. In previous versions of XDPROP,  $V(\mathbf{r})$  was computed using central contributions according to Su and Coppens [6], and peripheral contributions according the ADF method [7-9]. In the current version of XDPROP, the central contributions are now obtained by the method recently described by Volkov *et al* [56]. Thus the potential may be computed at points inside the charge distribution, permitting the calculation of  $V$ , the electric field vector **ef**, and the electric field gradient **efg** at the nuclei, and exploration of the topology of  $V(\mathbf{r})$  [39]. **NOTE:** Since the **efg** is not scalar or vector quantity, this property can only be calculated at a single point (*i.e.* properties in a line or 2D or 3D maps *cannot* be calculated).

The nuclear potential is just the contribution of the nuclei alone to (3.15), *i.e.*

$$V(\mathbf{r}) = \sum_j Z_j / |\mathbf{r} - \mathbf{R}_j| \quad (\text{Eq. 6-4})$$

**sigrho** and **siglap** are currently computed using only the contributions from population parameters and  $\kappa$ 's taken into account. Hence these should be considered lower-limits. Note that **sigrho** and **siglap** may be very tedious to compute, so you might want to think twice before trying to generate a cube of **siglap** data !

If flagged, the keyword **nci** calculates the reduced density gradient  $s(\mathbf{r})$  for displaying Non-Covalent Interactions (NCI) as proposed by Johnson *et al* [58].

$$s(\mathbf{r}) = |\nabla \rho(\mathbf{r})| / [2(3\pi^2)^{1/3} \rho(\mathbf{r})^{4/3}]$$

An isosurface plot for a low  $s(\mathbf{r})$  value and for low electron densities, mapped and colour coded with the electron density multiplied by the sign of the second eigenvalue of the Hessian (the signed density **srho**) provides an indication of the existence and nature of non-covalent interactions [58]. A grid file of this latter property can be computed by selecting the property **srho**. By examining plots of the  $s(\mathbf{r})$  versus electron density, a suitable value for the electron density cutoff parameters *rhocut1* and *rhocut2* (see below) may be determined and used for computing a 3D grid of **nci** values where only  $s(\mathbf{r})$  values for low electron density points associated with non-covalent interactions are significant. For the purpose of this initial plot, the two cutoffs should be set at their default values of zero in the calculation of the **nci** grid file, and identically sized gridfiles of **rho** and **srho** should also be calculated. Saleh *et al* [59] have proposed mapping the  $s(\mathbf{r})$  isosurface using the kinetic energy density  $G(\mathbf{r})$ , the potential energy density  $V(\mathbf{r})$  and total energy density  $H(\mathbf{r})$ . Grid files of these approximate properties may be created by selecting the properties **ked**, **ped** or **ted**. The Kirzhnits approximation [40] for  $G(\mathbf{r})$  used in computing these energy densities is only valid in regions of low electron density, so is a reasonable approximation for this purpose. The two grid files of the **nci** and the chosen mapping property must be of identical size and origin.

The keyword **dori** computes the Density Overlap Regions Indicator (DORI) as recently defined by de Silva & Corminboeuf [60]. This unitless function, with values between 0 and 1, is a new indicator which allows simultaneous visualization of both covalent and non-covalent interactions. Isosurfaces of the DORI, colour-coded either with the signed rho **srho** or the second eigenvalue of the Hessian (property **eig2**) may be interpreted in a similar fashion to the NCI. In addition the Single-Exponential Decay Detector (SEDD), low values of which show regions of spaces where the electron density decays in a single exponential manner, may be computed by selecting the property **sedd** [61]. The **dori** and **sedd** properties, which involve a calculation of a quotient, necessitate a check on the value of the denominator at all points. In regions of low density, the denominator tends to zero, in which case a default value for the property is assigned. In these regions, the minute errors in the

density and its derivatives can result in artefacts, so a non-zero cut-off can be applied using the flag **dncut** in the **SELECT** section (see below). The default value of this flag is zero, but a small non-zero value like  $10^{-15}$  could be tried if artefacts in low density regions in the plots are apparent. In XD, the derivatives of  $\rho$  are calculated analytically (unless otherwise requested by the user) so they are equally well defined, regardless of the magnitude of  $\rho$ . However, for small  $\rho$  the numerical instability of many functions like **dori**, **sedd** and **nci** becomes larger. In the multipolar expansion the relative errors of  $\rho$ ,  $\nabla(\rho)$  and  $\nabla^2(\rho)$  are larger when  $\rho$  is smaller, so the relative error produced by a fitting may be quite important in those function at smaller  $\rho$ . This just concerns the error of the fitting, without considering the propagation of the experimental error.

The one-electron-potential function **oep** [43-45] is defined as:

$$\text{OEP} = \nabla^2 \sqrt{\rho} / (2\sqrt{\rho}) = 1/4(\nabla^2 \rho / \rho) - 1/8(\nabla \rho / \rho)^2$$

Two other approximate properties, as proposed by Tsirelson, namely the Electron Localization Function **elf** [41a] and the Localized Orbital Locator **lol** [41b] may also be calculated. These are somewhat deprecated and not included in the standard `xd.mas` file, as both functions are computed using the approximation of Kirzhnits [40] for the kinetic energy density. This approximation is very poor in regions of high electron density, so that *it is completely inappropriate to analyse the topology of these scalar fields* using XDPROP. Moreover, the approximate **elf** has been shown by Tsirelson [41a] and Jayatilaka [42] to introduce undesirable artefacts.

#### 6.2.4 SELECT (\*)flag

This first type of SELECT command involves flags, namely **numdx**, **esd**, and **au**, which require no further qualifiers and **verbose**.

##### (\*)numdx

CP searching a function requires a full set of first and second derivatives with respect to a global cartesian coordinate system. Analytic derivatives are available for many functions, and for others the derivatives will be computed numerically with finite differences. Placing an asterisk by this keyword forces the program to use numerical derivatives for CP searching *any* function, regardless of whether analytic derivatives are available.

##### (\*)esd

The default is not to compute any errors since this needs the variance-covariance matrix file `xd.cov` which, due to its large size, may have been deleted prior to running XDPROP. Selecting **esd** turns on computation of errors, in so far as they are available for different properties. Currently errors are only available for: the dipole moment,  $\rho$  (or  $\Delta\rho$ ) and  $\nabla^2\rho$ . In least squares refinement, the variance in some property  $A$ , derived from the  $N_{\text{vars}}$  refined parameters  $\{P_i\}$ , may be estimated from the variance-covariance matrix  $\epsilon$  [5]:

$$\sigma^2(A) = S^2 \sum_{i,j}^{N_{\text{vars}}} \left( \frac{\partial A}{\partial P_i} \right) \epsilon_{ij} \left( \frac{\partial A}{\partial P_j} \right) \quad (\text{Eq. 6-5})$$

where  $S$  is the goodness-of-fit.

##### (\*)au

If selected, properties are output in atomic units. This DOES NOT affect the input parameters in `xd.mas` and `xd.inp` which are generally in Angstrom units. **Note** : not all program options accept this directive.

##### verbose verbose\_level

Setting `verbose_level` greater than one switches on extra printout to the screen and to the log files. The old flags **\*check** and **\*debug** will also still work.

**(\*)local**

This keyword is only mentioned because it was used in old versions of the program. It is no longer present in the default `xd.mas` file written by the current version of XDINI, and is ignored in the current version of XDPROP. The program action is always to use a set of multipole populations transformed into a *global coordinate system* for property computation. It is no longer possible to run calculations using the local coordinate systems.

## 6.3 Instructions to Define the Cluster of Atoms Used

### 6.3.1 APPLY

**APPLY** *symm* *is trans* *tx ty tz* (*atoms ...* | **all**)

This command is used to apply a crystallographic symmetry operation to some or all of the pseudoatoms in the asymmetric unit. This can be useful if, for example, the asymmetric unit does not contain a whole molecule, or if the properties of two or more molecules are to be analysed, or for studying intermolecular hydrogen bonding. The symmetry operations are referenced *according to the sequence in which they are listed in xd\_pro.out*. For example, the following entry

```
APPLY SYMM 3 TRANSLATIONS -2 -1 1 O(1) N(1) C(1) H(1) H(2) H(3)
```

applies the third symmetry operation, with extra lattice translations, to the six atoms whose labels are given. If you do not wish to supplement the symmetry operation with additional lattice translations, use `TRANSLATIONS 0 0 0`. If the symmetry operation is to be applied to *all* atoms in the asymmetric unit, then the keyword **all** may be used in place of the atom labels. More than one line beginning with **APPLY** may be present. There is a limit of a maximum of 7 atoms which may be specified individually on any single **APPLY** instruction.

One important point to note is the XDPROP convention for labelling symmetry-generated atoms. This label is of the general form "X"+sym.op number+"\_"+original atom label. So for example, an atom generated from O(8) by symmetry operation 6 will have the label X6\_O(8). This label should be used to refer to the symmetry generated atom in the other modules of XDPROP, for example if you want to use such an atom to define a plane, or in a CP search.

### 6.3.2 GROUP

**GROUP** (**not**) *atoms ...*

Whereas **APPLY** is concerned with expanding the number of pseudoatoms under study, **GROUP** has the opposite effect of selecting a subset of pseudoatoms to be studied. Henceforth all sums over pseudoatoms to compute properties will include only those pseudoatoms whose atom labels follow the **GROUP** keyword. Thus an example might be

```
GROUP H(1) O(1) H(2)
```

If the option **not** is given, all atoms specified in this **GROUP** instruction are *not* part of the active group.

```
GROUP NOT H(1) O(1) H(2)
```

The effect of two or more **GROUP** instructions is additive. It is possible to reset the active group using the keyword **ALL**. `GROUP NOT ALL` or `GROUP ALL`

There are effectively no lower or upper limits on how many atoms can make up the group. This command is useful, for example, when the asymmetric unit contains several moieties, and you want to compute the properties of just one of them. The negated form might be useful if you want to exclude only a small number of atoms.

### 6.3.3 DEFGROUP

**DEFGROUP** *atoms ...*

This is specialized option concerned only with the deformation density. It will have no effect if the active property is not DEFDEN. The purpose of DEFGROUP is to define a subset of atoms whose spherical atom density will *not* be subtracted from the total density in creating the deformation density. The normal option (the default) is to subtract the spherical atom density at the nuclear positions for all the atoms in the active group. The syntax is simply (for example)

```
DEFGROUP Ni(1) Ni(2)
```

### 6.3.4 OMIT

This instruction allows the user to exclude atoms from further calculations. Either an explicit or implicit list of atom serial numbers may be given. For example, the three **OMIT** instructions shown below all have the same result of excluding atoms 3 to 8.

```
OMIT 3 4 5 6 7 8
OMIT 3 -8
OMIT 3 -5 6 -8
```

### 6.3.5 CRYSTAL

**CRYSTAL** **alim** *xmin xmax* **blim** *ymin ymax* **clim** *zmin zmax*

This directive adds in all atoms contained in the volume defined by the unit cell vectors **alim**, **blim** and **clim**. *xmin xmax* are the minimum and maximum allowed fractional coordinates of atoms along the unit cell axis **a** for generating of symmetry equivalent molecules (atoms). The default is -1.0 2.0, *i.e.* only atoms with  $-1. < \mathbf{x} < 2.$  will be generated. *ymin ymax zmin zmax* are similarly defined along the unit cell axes **b** and **c** respectively and have the same defaults.

## 6.4 Parameters for Topological Analysis

### 6.4.1 SELECT

**SELECT** (*keyword value*) ...

So far as is possible, all parameters in XDPROP have some useful default values. However, you will doubtless find it necessary to change something, and a number of parameters may be altered using this option. More than one card beginning with **SELECT** may be present, and each may contain up to five of the keywords to be described shortly. The sub-options correspond to system variables, and the numerical value of the variable must follow the keyword.

**cpcut** *dxcut* [**1.0e-6**]

This is used in CP searching. A CP is considered to have been located at some point  $\mathbf{r}_c$  if  $|\nabla f(\mathbf{r}_c)| < dxcut$ . The default value is  $1.0 \times 10^{-6}$  Å, which is generally fine for  $\rho$ , but may be quite inappropriate for searching other functions.

**lmax** *lmax* [**4**]

This is a global maximum *l*-value applied to all pseudoatoms. The default procedure is to carry out multipole sums on each pseudoatom to the maximum *l*-value in place on that atom. This will be overridden by the **lmax** option (an integer variable must be given). Although it is not possible to change *lmax* on individual pseudoatoms using these master file instructions, it could of course be achieved by manually editing the `xd.res` file.

**nstep** *nstep* [20]

Critical Points (CPs) are located by an iterative procedure - this parameter determines the maximum number of steps used in searching for a CP, before the search is abandoned.

**rcut** *rcut* [4.0]

Local properties  $f(\mathbf{r})$  are computed as a sum over pseudoatom contributions:  $f(\mathbf{r}) = \sum_j f_j(\mathbf{r})$ . For many properties, notably  $\rho$ ,  $\nabla^2\rho$  and  $\Delta\rho$ , it is an excellent approximation to ignore contributions of pseudoatoms which lie further than a distance *rcut* from the point  $r$ . The default distance of 4.0 Å is generally reliable for the afore-mentioned properties, and organic compounds. It is not likely to be suitable (i) when 'large' atoms are present, e.g. transition metals (ii) for  $V(\mathbf{r})$  or  $\nabla(\mathbf{r})$  (iii) when some pseudoatoms have rather small values of  $\kappa$  and/or  $\kappa'$ . *Users are therefore urged to check that the property being computed is converged with respect to this parameter.*

When the properties *esp*, *ef* or *efg* are computed, the parameter **rcut** has a rather different action. At the distance specified by **rcut**, the calculation switches from the exact to an approximate formula, and for this purpose, the value of **rcut** should be increased. For lighter elements a value of 6.0 Å has given good results, and an even larger value may be appropriate for heavier elements such as the transition metals.

**scale** *scale* [0.05]

CP searching involves consecutive steps  $\delta s$  towards stationary points. The direction of each step is determined by the local gradient vector and Hessian matrix for the property concerned. The step-length, on the other hand, is controlled by the variable *scale*, with a default of 0.05 Å. This value works well for searching  $\rho$ , but other (most likely smaller) values should be considered in searching more rapidly-changing functions such as  $\nabla^2\rho$ .

**dx** *xstep* [0.001]

The step-length for numerical derivatives (used in CP searching those properties for which analytical derivatives are not yet available).

**ds** *pstep* [0.005]

Bond paths are determined by numerical integration of the gradient vector  $\nabla\rho(\mathbf{r})$  along the line of maximum density, and *pstep* is the (initial) step-length for this integration.

The following two options apply to the calculation of the electrostatic potential (**esp**) only:

**rEcrit** *rEcrit* [0.00001]

*rEcrit* is the critical distance from the nucleus at which the calculation of the electronic potential switches from the "central" method of Volkov *et al* [56] to the "peripheral" ADF method [7-9].

**rNcrit** *rNcrit* [0.00001]

*rNcrit* is the critical distance to a nucleus, below which the nuclear potential is omitted entirely.

**dncut** *dncut* [0.0]

*dncut* is used in the calculation of the several properties, including **nci**, **dori** and **sedd** to avoid zero divide errors. The calculation of these properties is only made in cases where the denominator involving a function of  $\rho$  is greater than the value of *dncut*, otherwise default values are assigned to the property. A small positive non-zero value (e.g.  $10^{-15}$ ) may be tried to remove the artefacts in these properties which may appear in regions of very low electron density.



**rhocut1 rhocut1 [0.0] rhocut2 rhocut2 [0.0]**

*rhocut1* and *rhocut2* are used in the calculation of the properties **nci** and **dori**, to select ranges of the electron density  $\rho$  in which the property is examined. All points in the **nci** grid file where  $\rho$  is less than *rhocut1* or greater than *rhocut2* (where *rhocut2* is greater than zero) will be assigned an arbitrary large **nci** value, so that these points will not be visible in isosurface plots of low **nci** values. Similarly for **dori** grid files, the **dori** value will be assigned as zero. With the default value of zero for *rhocut2*, no upper cut-off is applied. Suitable values of *rhocut1* and *rhocut2* for **nci** can be determined by examining plots of the reduced density gradient versus  $\rho$ , for the system under consideration. Typical values for the two parameters could be 0.0 and 0.2, as the low **nci** values associated with covalent bond critical points generally have  $\rho$  values around 1.0 or higher. For transition metal complexes, the value of  $\rho$  at the bcp for M-ligand bonds is often less than 1.0, so a careful examination will be necessary in these cases to isolate the non-covalent interactions.

**6.4.2 QUADINT****QUADINT iqt iqt Nrad Nrad Nang Nang (\*)Becke (\*)Stock**

This instruction controls the quadrature integration procedures for the stockholder moments (**STOCKMOM**), the interaction energies (**INTEREN**), the lattice energy (**LATEN**) and the atomic basin integrations (**TOPINT**) calculations.

**iqt iqt [2]**

*iqt* defines the type of radial quadrature used. If *iqt* = 1, the Gauss-Legendre quadrature with Becke's mapping [10] is used. If *iqt* = 2, the Gauss-Chebyshev quadrature [11] with the Treutler-Ahlrichs M4 mapping [12] is used. If *iqt* = 3, the Euler-MacLaurin quadrature [13] with the Mura-Knowles mapping [14] is used.

**Nrad Nrad [50]**

*Nrad* is the number of radial points used.

**Nang Nang [194]**

*Nang* is the number of angular points used. The Lebedev-Laikov quadrature [15] is used for the angular integration. Almost all of the Lebedev quadratures are available, allowed values of *Nang* range from 6 to 5810. The user may enter any value for *Nang* - the program automatically assigns the closest allowed Lebedev quadrature to the input value.

Weighting functions of the Becke form [10] (**\*Becke**) or the stockholder form [16] (**\*Stock**) can be used to partition the 3-D space. These can be both switched off, for example for pseudoatom-pseudoatom integration.

**6.4.3 ODESOLVE****ODESOLVE \*rk (\*)bs eps eps stepi stepi**

This instruction controls parameters for the ODE (Ordinary Differential Equation) solver used in the integration procedures for the **TRAJPLT** (Section 6.6.6) and **TOPINT** (Section 6.8) options. Two algorithms are available, the fifth-order Runge-Kutta (**\*rk**) and the Bulirsch-Stoer (**\*bs**). Both methods use monitoring of the local truncation error to ensure accuracy and automatic adjustment of the stepsize.

**eps eps [1.0d-6]**

The required accuracy of integration

**stepi stepi [0.01]**

The initial stepsize in Angstroms

**6.4.4 ORIGIN****ORIGIN \*mass (\*)charge (\*)geom (\*)ucell**

The instruction defines the origin used in the calculation of molecular moments. This will affect ALL instructions which calculate molecular moments, *i.e.* **LATEN**, **INTEREN**, **STOCKMOM**, **MULTMOM** and **TOPINT**. One of four frames may be specified : **mass** uses the center of mass as the origin, **geom** the geometric center, **ucell** the origin of the unit cell and **charge** the center of charge distribution (*i.e*  $Z$ ). The origin so defined remains effective until another **ORIGIN** instruction is given.

**6.4.5 EXPORT****EXPORT \*orient \*min16 lmax lmax nmol nmol natmol n1 n2 ...**

This instruction, in conjunction with the directives **MULTMOM**, **STOCKMOM** or **TOPINT**, causes XDPROP to write out atomic moments in spherical tensor notation [21] in the input formats for the programs ORIENT [22] (when **orient** is flagged) and/or MIN16 [23] (when **min16** is flagged). *lmax* defines the maximum level of atomic moments to be written. *nmol* defines the "number of molecules" present in the atom list (including atoms generated with **APPLY** and **GROUP** instructions). *n1*, *n2* .. define the number of atoms in each of the molecules. By default there is only 1 molecule, which includes all atoms in the list. The **EXPORT** instruction MUST precede any **MULTMOM**, **STOCKMOM** or **TOPINT** directives.

**6.5 Calculation of Properties****6.5.1 MULTMOM**

The traceless Cartesian molecular moments up to the hexadecapole level are calculated from the multipole populations. This option replaces the previous **DIPOLE**, **QUADPOLE** and **MOLMOM** directives, which are now aliases for **MULTMOM**. The molecular moments can be calculated in one of four frames dependent on the **ORIGIN** directive (see Section 6.4.4).

The user is referred to the excellent review by Spackman on computing molecular multipole moments from X-ray diffraction data [17]. The computation of the molecular dipole moment requires some definition of a "molecule" in the crystal [18-20] since the positioning of the intermolecular boundary is arbitrary. XDPROP assumes that the density of a molecular fragment in the crystal is derivable from just the *n* pseudoatoms that would describe the free molecule. This ignores any overlap between pseudoatom multipole functions on neighbouring molecules. Then the dipole moment is given by

$$\mathbf{p} = \sum_{j=1}^n \mathbf{p}_j \quad (\text{Eq. 6-6})$$

where

$$\mathbf{p}_j = Z_j \mathbf{R}_j - \int \mathbf{r} \rho_j(\mathbf{r}_j) d\mathbf{r} \quad (\text{Eq. 6-7})$$

$\mathbf{R}_j$  is the nuclear position vector, and  $\mathbf{r}_j = \mathbf{r} - \mathbf{R}_j$ . In a monopole-only refinement, the pseudoatom dipole contribution is given by

$$\mathbf{p}_j = Z_j \mathbf{R}_j - \int \rho_j(\mathbf{r}_j)(\mathbf{r}_j + \mathbf{R}_j) d\mathbf{r} = q_j \mathbf{R}_j \quad (\text{Eq. 6-8})$$

which follows from the normalization of the monopole functions ( $\int \rho_j(\mathbf{r}_j) d\mathbf{r}_j = 1$ ), and the odd symmetry of the second integrand ( $\int \rho_j(\mathbf{r}_j) \mathbf{r}_j d\mathbf{r}_j = 0$ ). Only the monopoles and dipoles contribute to  $\mathbf{p}_j$  due to the symmetries of the  $d_{lm}$  and the dipole operator. If the radial functions are normalized Slater-type functions

$$R_l(r) = \frac{\zeta_l^{n+3}}{(n+2)!} r^{n_l} e^{-\zeta_l r} \quad (\text{Eq. 6-9})$$

then the remaining integrations may be performed analytically, to give

$$\begin{aligned} \mathbf{p}_j &= q_j \mathbf{R}_j - P_{1,+1} \int d_{1,+1} R_l x d\mathbf{r} \\ &\quad - P_{1,-1} \int d_{1,-1} R_l y d\mathbf{r} - P_{1,0} \int d_{1,0} R_l z d\mathbf{r} \\ &= q_j \mathbf{R}_j - \frac{4\mathbf{P}'_j(n_l+3)!}{3\kappa_j'' \zeta_j(n_l+2)!} \end{aligned} \quad (\text{Eq. 6-10})$$

where  $\mathbf{P}'_j$  is the vector consisting of the dipole populations for the  $j^{\text{th}}$  atom. In general, each atom has its own local coordinate system so  $\mathbf{P}_j$  must include the effect of a rotation  $\mathbf{A}$  to a common Cartesian frame:  $\mathbf{P}'_j = \mathbf{A}_j \mathbf{P}_j$ .

The dipole moment  $\mathbf{p}$  is independent of the choice of origin (for a neutral moiety), but the dipole moment variance  $\sigma(\mathbf{p})$  would normally be origin-dependent. However, the application of the neutrality constraint in XDLSM has an effect equivalent to minimizing the dipole variance with respect to the origin. Therefore molecular or fragment dipole moment variances computed with XDPROP should be origin-independent, *provided that the neutrality constraint was applied appropriately during refinement*.

## 6.5.2 D-POP

### D-POP use \*Pv \*P00

Occupancies of the  $d$  orbitals are derived, (with the assumption of low  $sp$  to  $d$  overlap) according to the approach suggested by Holladay, Leung and Coppens [32]. The calculated  $d$  orbital occupancies are given in the output file `xd_pro.out`. Either or both of the populations in Pv and P00 may be used in the calculation. If the **\*esd** flag in the **SELECT** heading is set, estimated errors are also tabulated. For example:

```
MULTIPOLE POPULATIONS
P00
2.376
P20  P21+  P21-  P22+  P22-
0.094 0.000 0.000 0.000 0.000
P40  P41+  P41-  P42+  P42-  P43+  P43-  P44+  P44-
0.213 0.000 0.000 0.000 0.000 0.001 0.011 0.000 0.000
ORBITAL POPULATIONS
z2      = 0.87065
xz      = 0.32579
yz      = 0.32579
x2-y2   = 0.42701
xy      = 0.42701
z2/xz   = 0.00000
z2/yz   = 0.00000
z2/x2-y2 = 0.00000
z2/xy   = 0.00000
xz/yz   = 0.00000
xz/x2-y2 = 0.00126
xz/xy   = 0.02387
yz/x2-y2 = 0.02387
```

```
yz/xy    = -0.00126
x2-y2/xy = 0.00000
tot d-pop = 2.37625
```

### 6.5.3 STOCKMOM

Quadrature integration of the electron density, partitioned according to the Stockholder method [16], and calculation of the corresponding unabridged moments is performed. The relevant instructions in the `xd.mas` file are:

```
QUADINT iqt iqt Nrad Nrad Nang Nang (*)Becke (*)Stock
STOCKMOM atoms (*)all *unique (*)select ato(1) ato(2) ...
```

For the **QUADINT** instruction, see Section 6.4.2

```
atoms (*)all *unique (*)select ato(1) ato(2) ...
```

If **select** is flagged, only the specified atoms `ato(1) ato(2) ...` will be integrated, if **all** is flagged all atoms in the atom list will be integrated, or if **unique** is flagged then only the atoms in the asymmetric unit will be integrated (default).

### 6.5.4 INTEREN

Interaction energies between two molecular fragments or atoms are calculated by numerical (quadrature) integration and/or by the Buckingham approximation using either atomic or total molecular multipolar moments, supplemented with various atom-atom potentials for calculation of the exchange-repulsion, dispersion and induction energies. The total interaction energy may be expressed [21] as the sum of electrostatic, exchange-repulsion, dispersion and induction terms.

$$E_{\text{int}} = E_{\text{es}} + E_{\text{ex-rep}} + E_{\text{disp}} + E_{\text{ind}}$$

The electrostatic term  $E_{\text{es}}$  is usually the major contribution to  $E_{\text{int}}$ . The exchange-repulsion term arises from the antisymmetrisation requirements of the Pauli principle, the dispersion term from the induced charge distributions on different fragments, and the induction term from the interaction of the unperturbed charge density on one fragment with the induced charge distribution another (and vice-versa). In XDPROP, the latter two terms are obtained using empirically derived atom-atom potentials. The induction term is not normally required for calculations involving experimental data, since the induction effects of the crystal lattice should already be included in the experimentally determined pseudoatom parameters. While several potentials are available, the user should note that none have been proven reliable for all classes of compounds in all types of molecular environment. In the current version of XDPROP, it is assumed that the atoms comprising the two interacting fragments are entered consecutively in the `xd.mas` file. The following instructions are relevant:

```
QUADINT iqt iqt Nrad Nrad Nang Nang (*)Becke (*)Stock
ATATPOT *exrep (*)spack *wilcox (*)lj
ATATPOT *dispr (*)spack *wilcox (*)lj
HBONDS Hatom1 Xatom1 Distance1, (Hatom2 Xatom2 Distance2 ..... )
HPOLAR Hatom1, Hatom2 .....
INTEREN frag 1 nat1 -nat2 (*)neutral
INTEREN frag 2 nat3 -nat4 (*)neutral
INTEREN (*)EP (*)aMM (*)mMM (*)EPMM rCrit1 rCrit1 rCrit2 rCrit2
```

The **ATATPOT** instructions are required if calculation of the exchange-repulsion, dispersion and induction terms is to be included. The **HBONDS/HPOLAR** instructions might also be required in the presence of H-bonds.

For parameters controlling the quadrature integration **QUADINT**, see Section 6.4.2 The instructions **ATATPOT/HBONDS/HPOLAR** are common to both the **INTEREN** and **LATEN** calculations and are described first.

**IMPORTANT :** Atom-atom parameters are currently available only for a limited number of atomic types. If the structure contains other elements, only the electrostatic term of the interaction energy can sensibly be computed. Contributions to the exchange-repulsion, dispersion and induction terms from any atomic type with unavailable parameters are set to zero !

The **ATATPOT/HBONDS/HPOLAR** instructions **MUST** precede any **INTEREN** or **LATEN** instructions in the `xd.mas` file.

#### **ATATPOT \*exrep (\*)spack \*wilcox (\*)lj**

This instruction specifies the type of atom-atom potentials to be used for the *exchange-repulsion* term of the total energy. The instruction is only executed if the **exrep** parameter is selected (*i.e.* starred).

|               |   |
|---------------|---|
| <b>spack</b>  | Spackman's potentials for exchange-repulsion and dispersion [24].<br>Available for elements H, C-F, Al-Cl, Ga-Br. |
| <b>wilcox</b> | Williams and Cox potentials for exchange-repulsion and dispersion [25].<br>Available for elements H, C, N, O      |
| <b>lj</b>     | Pairwise Lennard-Jones 12-6 (and 12-10 for H-bonds) potentials [26]<br>Available for elements H, C, N, O          |

#### **ATATPOT \*dispr (\*)spack \*wilcox (\*)lj**

This instruction specifies the type of atom-atom potentials to be used for the *dispersion* term of the total energy. The allowed options are the same as for the exchange-repulsion instruction. The instruction is only executed if the **dispr** parameter is selected (*i.e.* starred).

**IMPORTANT :** Since Spackman's potentials for exchange-repulsion contain a spherical-density/spherical-density term, these potentials **SHOULD NOT** be used in conjunction with the EPMM method. The latter already includes this term, and using both will result in an overestimation of exchange-repulsion. The Williams and Cox potentials for exchange-repulsion should be used in preference, as the Lennard-Jones potentials are very approximate, and acceptable only for very weak interactions.

Spackman's potentials for exchange-repulsion **COULD** be used with the **aMM** (atomic multipole moments) method, but this procedure is not very good for short contacts, although it might be acceptable for less tightly bound crystals. For the dispersion term either Spackman's or Williams and Cox potentials may be used with any method (they give generally similar results).

The same considerations also apply to the **LATEN** instruction.

### **HBONDS Hatom1 Xatom1 Distance1, ....**

This instruction specifies which hydrogen atoms are involved in H-bonds, and is used to set special potentials for those atoms. Xatom1 is the acceptor atom for set #1 of hydrogen bonds and Distance1 is the distance from the H atom to the acceptor atom. Up to 200 sets of H-bonds may be specified.

### **HPOLAR Hatom1, Hatom2 .....**

This instruction specifies which hydrogen atoms involved in H-bonds are to be considered as polar. Different potentials are used for such atoms.

### **INTEREN frag n n1 -n2 \*neutral**

This instruction defines the atomic fragment as a consecutive run between atoms *n1* to *n2*. The fragment number *n* may be either 1 or 2 - the program will halt if more than two fragments are defined. If the **neutral** directive is selected, the fragment will be made electroneutral.

### **INTEREN (\*)EP (\*)aMM (\*)mMM \*EPMM rCrit1 rCrit1 rCrit2 rCrit2**

**EP/aMM/mMM/EPMM** control the type of calculation used to compute the potential. Only **one** method should be selected. The EPMM hybrid method is a combination of the three methods and is the preferred approach. It is described in references [27,28], and has been shown [29] to provide accurate electrostatic interaction energies.

- \*EP**            The Exact Potential [30] is computed. **NOTE** : very compute intensive !!
- \*aMM**        The Buckingham approximation [31] using **atomic** moments is used.
- \*mMM**        The Buckingham approximation [32] using **molecular** moments is used.
- \*EPMM**      The EP/MM hybrid method [27,28] is used.

### **rCrit1 rCrit1 [4.0]**

The critical distance (Angstroms) at which the **EPMM** calculation switches between **EP** and **aMM**

### **rCrit2 rCrit2 [15.0]**

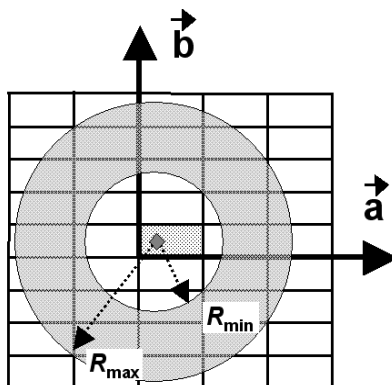
The critical distance at which the **EPMM** calculation switches to **mMM**.

If *verbose\_level* (Section 6.2.4) is greater than 1, then additional information is printed to Fortran units 48 and 49.

**IMPORTANT:** When using EP method, one should always check the accuracy of integration by comparing numerical values for two integrals: integral of  $\rho(A) \cdot \text{elpot}(B)$  and integral of  $\rho(B) \cdot \text{elpot}(A)$  where A and B are the pair of pseudoatoms between which Ees is evaluated. For more information, see [28b], equation 3 and discussion therein. This can be done by setting 'verbose 3'. In this mode, XDPROP will create two files: fort.48 and fort.49 In file fort.49, the values of the two integrals mentioned above are compared. For each pair of pseudoatoms integrated, check the differences between the two integrals listed in the fort.49 file. If it is too large even for one pair, simply increase the accuracy of integration by giving larger values for Nrad and Nang in QUADINT, for example, going from 50x194 to something like 70x194, or 70x302, 100x302 etc, until you get an acceptable agreement between the two integrals for ALL pseudoatom pairs.

### 6.5.5 LATEN

The electrostatic crystal binding energy is calculated using the same methodology as described above for **INTEREN**. All intermolecular interactions are calculated directly within a spherical shell centred about the reference molecule in the asymmetric unit. The radii of the internal and external "surfaces" of the shell are defined by  $R_{min}$  and  $R_{max}$ .



The following instructions are relevant in the `xd.mas` file:

```
QUADINT igt igt Nrad Nrad Nang Nang (*)Becke (*)Stock
ATATPOT *exrep (*)spack *wilcox (*)lj
ATATPOT *dispr (*)spack *wilcox (*)lj
HBONDS Hatom1 Xatom1 Distance1 (Hatom2 Xatom2 Distance2 ..... )
HPOLAR Hatom1, Hatom2 .....
LATEN frag1 nat1 -nat2 *neutral
LATEN radii Rmin Rmax
LATEN (*)EP *aMM (*)mMM (*)EPMM rCrit1 rCrit1 rCrit2 rCrit2
```

For **QUADINT**, see Section 6.4.2. For **ATATPOT/HBONDS/HPOLAR** instructions see Section 6.5.4.

**LATEN frag1 nat1 -nat2 \*neutral**

Defines the fragment to be calculated, from *nat1* to *nat2*. If the **neutral** directive is selected, the fragment will be made electroneutral.

**LATEN radii rad1 rad2**  
**radii rad1 rad2 [0.0, 100.0]**

These provide a definition of the shells, in terms of radii in Angstroms.

**LATEN (\*)EP \*aMM (\*)mMM (\*)EPMM rCrit1 rCrit1 rCrit2 rCrit2**

**EP/aMM/mMM/EPMM** are directives controlling the type of calculation undertaken (see corresponding **INTEREN** instruction for details). Only **one** should be selected.

**rCrit1 rCrit1 [4.0]**

The critical distance (Angstroms) at which the **EPMM** calculation switches between **EP** and **aMM**

**rCrit2 rCrit2 [15.0]**

The critical distance at which the **EPMM** calculation switches to **mMM**.

If *verbose\_level* is greater than 1 (Section 6.2.4), then additional information is printed to Fortran units 48 and 49.

### 6.5.6 QFIT

**QFIT** *grid npts length width constrain (true|false)*

**CONSTRAIN** *keys ...*

This activates the potential-derived (PD) charges algorithm, which follows the procedure described by Williams & Cox [48,49]. A cubic grid of (*npts*)<sup>3</sup> points is centered at the centroid of coordinates of the active molecule. This grid is of size *length* Å.  $V(\mathbf{r})$  is computed over this grid, and then a subset of these points in a shell of thickness *width* Å is selected for the fit. This corresponds to a shell of points, whose inner surface is taken as the Van der Waals radius of the nearest atom, plus the Van der Waals radius of hydrogen. The residual

$$\chi^2 = \sum_i^m \omega_i \left[ V_i^o - \sum_j^{n-1} q_j r_{ij}^{-1} + \left( \sum_j^{n-1} q_j - Z \right) r_{in}^{-1} \right]^2 \quad (\text{Eq. 6-11})$$

(*m* is the number of grid points used in the fit;  $V_o$  is the exact potential from the multipole model at the *i*-th grid-point;  $q_j$  is the PD charge for the *j*-th nucleus;  $r_{ij}$  is the distance from the *j*-th charge to the *i*-th grid point;  $Z$  is the net charge on the molecule; and  $\omega_i$  is the weight for the *i*-th point) is minimized by a least squares fit. In this version of **QFIT**, all grid-points have unit weights. The root-mean square fit parameter

$$\sigma = \left[ \frac{1}{m} \sum_i^m (V_i^o - V_i^{\text{model}})^2 \right]^{1/2} \quad (\text{Eq. 6-12})$$

is reported. Chemical symmetry constraints may be applied to the fitted charges. A full example is:

```
QFIT  grid 11  length 7.0  width 1.0  constrain true
CONSTRAIN  1 2 3 4 4 4
```

If *constrain* is **false** then the **CONSTRAIN** line need not be present. The example above, which refers to a six-atom molecule, will fit independent charges for the first three atoms, and then a single (constrained) PD charges. This might be used e.g. in formamide where the last three atoms are hydrogens. A second example is for acetamide, where the amine hydrogens (atoms 4 & 5) and methyl hydrogens (atoms 6-8) could be constrained to have equal PD charges within each group:

```
CONSTRAIN  1 2 3 4 4 5 5 5
```

It should be noted that there are many different ideas and algorithms for obtaining PD charges, and the routine in XDPROP is one of the more primitive. Algorithms which additionally constrain the point-charge dipole moment to reproduce some ‘exact’ moment have been available for some time [50]. A recent idea, based on reproducing  $V(\mathbf{r})$  at the *nuclear* sites, rather than in a volume around the molecule, has been developed by Su [51]. This gives charges which have some internal chemical significance for the molecule, rather than optimally predicting its intermolecular interactions.

### 6.5.7 NUCPROP (\*)all (\*)select ato1, ato1 -ato3...

This command invokes the calculation of the esp, ef and efg at the nuclear positions. The exact properties calculated depend on the specification of the PROPERTY directive (see **6.2.3**). If the property selected is esp, then only the esp is calculated. If the property selected is ef, then the esp and ef are calculated, while if the property selected is efg, then all three are calculated. By default (*i.e.* if any other property is selected) then all three are calculated.



**6.5.8 EPFG EP aMM mMM \*EPMM rCrit1 rCrit2 rCrit2**

This command controls the calculation of the esp, ef and efg. It is needed because their evaluations via the exact formula of Volkov *et al* [56] are very slow if a lot of atoms are present in the list, *i.e.* if the **LATEN** or **CRYSTAL** options are used.

**EP** - the exact formula of Volkov *et al* [56]

**aMM** - using atomic multipoles

**mMM** - using molecular multipoles (this approximation is only available for **LATEN** - in the calculation of the esp, ef and efg for plots and properties at the nucleus, only the **aMM** approximation is available)

**EPMM** - hybrid method (**EP/aMM/mMM** in **LATEN**, **EP/aMM** everywhere else)

**rCrit1** and **rCrit2** are for **EPMM** only

**rCrit1** - interatomic distance for switching from **EP** to **aMM**, default 6 Å

**rCrit2** - interatomic distance for switching from **EP/aMM** to **mMM** (**LATEN** only), default 15 Å

**6.6 Function Plots**

The following options are for computing values of the active property (*i.e.* flagged on the **PROPERTY** card) at specified points, along lines, or over 2-D and 3-D grids of points. An unlimited number of these instructions may be present. However, depending on the computer operating system, the various files which are produced may overwrite one another. Thus it may be necessary to re-run XDPROP every time you want to produce a new map file, or rename the files between runs.

**6.6.1 POINT (property at a point)**

**POINT** *x y z*

This keyword followed by three coordinates in the global frame specifies a point at which the active property will be computed. No map file is created by this option, the result only appears in `xd_pro.out`.

**6.6.2 LINE (property along a line)**

**LINE** *atom1 atom2 npts npts*

This option computes the value of the active property along a line between two nuclei, specified by their atom labels. A final parameter which must be supplied is the number of (equidistant) points at which the property will be computed. So an example could be

```
LINE C(1) O(2) NPTS 21
```

Alternatively, the user may request a property along a line defined by points.

```
LINE POINTS x1 y1 z1 x2 y2 z2 npts 50
```

Note that *x1,y1,z1* and *x2,y2,z2* are the global Cartesian coordinates of the two points.

**6.6.3 VECTOR PLOTS**

If a 2D or 3D *vector map* of the vector property electric field (ef) is required, an extra directive must be given *before* the MAP or CUBE directives. In this case, as well as the standard format XD grid file (`xd_<prop>.grd`) containing the *magnitude* of the vector (which can be

visualised using XDGRAPH), an MTVDAT file called `xd_<prop>.dat` is also produced. MTVDAT files may be visualised using the program **PLOTMTV** (for Linux versions see <http://rpmfind.net>). The program is invoked with the command line :

```
plotmtv -colorps -plotall xd_<prop>.dat
```

**VECPLOT** (\*)**labels** (\*)**zcut** *zcut* **omitrad** *omitrad* (\*)**vcut** *vcut* **vscale** *vscale* **xytol** *xytol* (\*)**dp** *dipole\_scale*

If **\*labels** is selected, atomic labels are included in the MTVDAT file

(\*)**zcut** *zcut* [**1.2**]

The range in Å along the z-axis of the plot to include atoms.

**omitrad** *omitrad* [**0.2**]

Omit grid points within *omitrad* around each nucleus

(\*)**vcut** *vcut* [**0.8**]

Do not plot vectors with magnitude > *vcut*. Units depend on the property.

**vscale** *vscale* [**1.0**]

Scale factor for vectors in the plot

**xytol** *xytol* [**0.4**]

Tolerance in the xy-direction of the 2-D plot to include atoms.

(\*)**dipole** *dipole\_scale*

Starring this option will plot a projection of the dipole moment vector. The number *dipole\_scale* is a scale factor, different from the scaling of vectors *vscale*.

#### 6.6.4 MAP (property over a 2-D grid of points)

**MAP atoms** *atom1 atom2 atom3 npts npts step step* (\*)**trans** *tx ty tz*

**MAP bvect1** *x1 y1 z1 bvect2 x2 y2 z2 cen x0 y0 z0 npts npts step step*

There are two ways of specifying the plane with **MAP**. The first uses three atom labels, and the map center is the centroid of the three atoms. The grid will be square (**npts** × **npts**), with a gridspacing of **step** in Å. You can also specify a translation of the origin, with respect to the grid axes. Translations of 1.0, 0.0, 0.0 or 0.0, 1.0, 0.0 will shift the origin 1 Å along the *x* or *y* grid axis directions (+x leftwards in the horizontal, +y downwards in the vertical directions), respectively. The **trans** directive *must* be starred to effect this. A translation of 0.0, 0.0, 1.0 selects the plane parallel to and 1 Å above the plane containing the three atoms (change the sign of the *z* translation to select planes *below* the plane of the atoms). The shifts in *x*, *y* and *z* can be non-integral. Thus an example is:

```
MAP ATOMS C(1) C(2) O(1) NPTS 61 STEP 0.1 TRANS 0.5 0. 0.
```

The second method is to supply two basis vectors, in the global Cartesian coordinate system. The map center must be given in the global Cartesian frame. **npts** and **step** have the same meaning as before. A complete example is:

```
MAP BVECT1 1. .0 .0 BVECT2 .0 .0 1. CEN 1.3 .2 -.1 NPTS 13 STEP 0.2
```

A map file is produced with the filename structure `xd_mid_<prop>.grd`. The maximum value of **npts** is limited only by the system memory available.

#### 6.6.5 CUBE (property over a 3-D grid of points)

**CUBE center** *x0 y0 z0 npts npts step step*

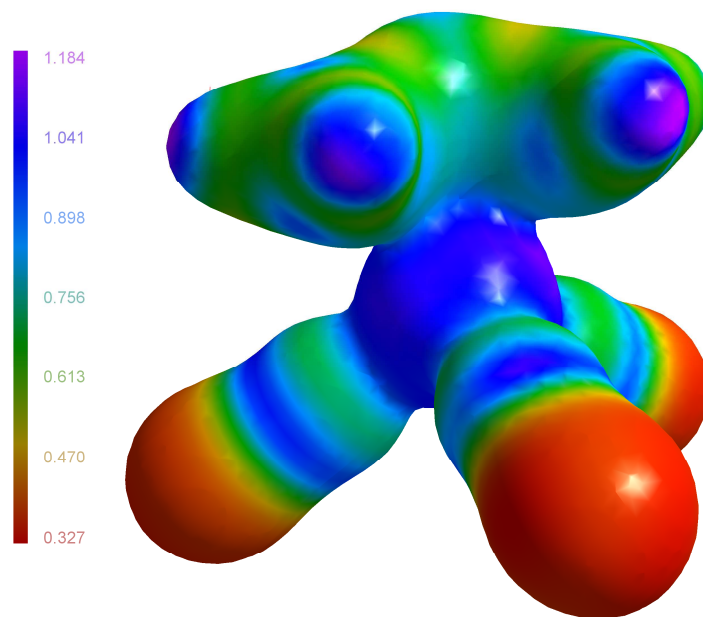
**CUBE** *atom1 atom2 npts npts step step*

This generates an  $npts \times npts \times npts$  cube of the active property, with a grid spacing of *step* (in Å). There are two ways of specifying the cube center: in global Cartesian coordinates, or as the midpoint of two atoms. Thus two examples are:

```
CUBE CENTER 0.3 9.82 1.5 NPTS 21 STEP 0.3
```

```
CUBE C(1) O(3) NPTS 21 STEP 0.3
```

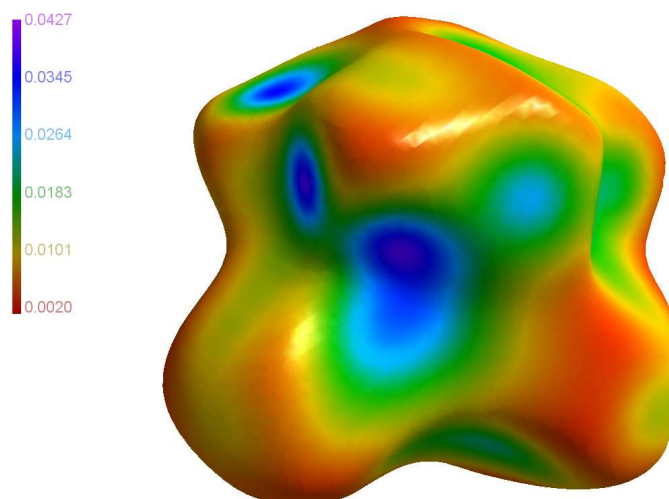
A map file is produced with the filename structure *xd\_mid\_prop.grd*. The maximum value of *npts* is limited only by the system memory available. A new feature of XD is that, in addition to the standard format XD grid files, the MAP and CUBE instructions now also generate CUBE files in Gaussian format. These may be used as input for programs like MOLEKEL v4.3 [54] or MoleCoolQT, so that one property, such as the electrostatic potential, may be mapped onto an isosurface of another property such as the density, as shown in Figure 6-1.



**Fig 6-1** Illustrative plot of isosurface mapping. The electrostatic potential is mapped onto an isosurface of the density (0.5 eÅ) in Cr(benzene)(CO)<sub>3</sub>. Red indicates the smallest (most negative) value of the potential and purple the largest (most positive). This plot was produced using the program MoleCoolQT from the Gaussian format CUBE files *xd\_rho.cube* & *xd\_esp.cube* output by XDPROP.

**CUBE** *hirshf* <*xd\_iam\_mol.grd*> <*xd\_iam\_cry.grd*>

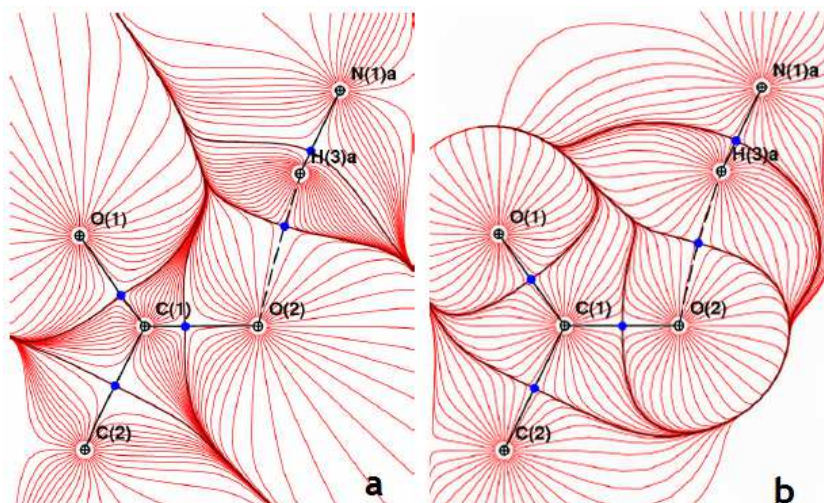
This is a special option to create a cube file with Hirshfeld weights in XD format, which can then be used to draw properties mapped onto the so called Hirshfeld surface [55]. Two cube files in XD format must be prepared previously, using the \*IAM option in XDPROP to create promolecule/procrystal densities. These cube file must have identical origins and dimensions. The first cube file *xd\_iam\_mol.grd* is a rho grid file of the isolated molecule, and the second cube file *xd\_iam\_cry.grd* is a rho grid file of the molecule in a crystal, i.e. completely surrounded by other molecules. This is best achieved using the **CRYSTAL** option (6.3.5) to expand the atom list. Cube calculations of this type are very time consuming, so some care should be taken to use the minimum expansion that is adequate. A new cube file *xd\_hirsh.grd* is created, which contains the ratio of the molecular:crystal density at all points. The Hirshfeld surface is the isosurface where this value is 0.5. This cube file may then be used as the iso-grid file with programs such as MoleCoolQt to map properties from a third cube file (Figure 6-2). This property cube file must again have the same origin and dimensions as the previous cube files.



**Fig 6-2** Illustrative plot of isosurface mapping. The electrostatic potential is mapped onto the molecular Hirshfeld surface of  $\text{Fe}(\text{TMM})(\text{CO})_3$ , using MoleCoolQt.

### 6.6.6 TRAJPLT

Plots of the gradient field trajectories of either the total electron density  $\rho$  or the electrostatic potential (esp) may be obtained. The plot is written to a PostScript file `<prop>_traj.ps`, which may be visualised with the GhostView program (available on Linux & Windows platforms). In addition, a Computer Graphics Metafile `<prop>_traj.cgm` is produced, which may be manually edited and converted to a PostScript file using the utility program RALCGM, which is available from the official XD website as a Linux RPM or Windows distribution.



**Fig 6-3** Illustrative TRAJPLT gradient-field trajectory plots of (a) the density and (b) the electrostatic potential of an alanine dimer in the plane of the carboxylate group.

The relevant instructions in the `xd.mas` file are :

**ODESOLVE \*rk (\*)bs eps eps stepi stepi**

See Section 6.4.3 for more details. The default value of `eps` may not be suitable for TRAJPLT plots. Smaller values of `eps` give smoother gradients vector lines, but increase the time taken. Values between  $1.0 \times 10^{-6}$  and  $1.0 \times 10^{-8}$  should give suitable plots for publication purposes. The fifth-order Runge-Kutta algorithm (**\*rk**) generally gives better results.

**TRAJPLT origin atom** *ato1*

*ato1* is an atom name which defines the origin of the plot.

**TRAJPLT origin point** <*x,y,z*>

Alternatively the origin may be given with a point, using global Cartesian coordinates (Å).

**TRAJPLT xaxis atom** *ato2* **xdim1** *xdim1* **xdim2** *xdim2*

*ato2* is an atom name which defines the x-axis of the plot, along the direction atom1-atom2. *xdim1* and *xdim2* are the minimum and maximum displacements of the plot from the origin

**TRAJPLT yaxis atom** *ato3* **ydim1** *ydim1* **ydim2** *ydim2*

The x-y plane of the plot is defined by *ato3*. The y-axis lies in this plane at 90 degrees to the x-axis. Alternatively, both the xaxis and yaxis definitions may also be given using **point** instead of **atom** as for the origin (see above).

**TRAJPLT mark** (\*)**atoms** (\*)**labels** (\*)**bonds** (\*)**cps** (\*)**basins** (\*)**hbonds**

This directive allows the user to select which of the graphical objects appear in the plot. Most are self evident - the **\*hbonds** directive results in long bonds (such as hydrogen bonds) being drawn as dashed lines, otherwise they will be drawn as a solid line, as 'normal' bonds.

**TRAJPLT params** **Circle** *rad1* **ATrad** *rad2* **CPrad** *rad3* **CPgrid** *step* **CPlim** *cplim*

This directive allows the user to modify the default values of the drawn radius of the circle around each atom from where the trajectories originate (*rad1*), the drawn radii of the atoms (*rad2*) and critical points (*rad3*). *step* is the grid size in Å for the CP search in the plot plane. (3,-1) critical points are drawn as blue circles, (3,+1) critical points as green circles and (3,-3) critical points as magenta circles. If the function value at a found critical point is lower than *cplim*, then that CP is ignored

**TRAJPLT** (\*)**plot** (\*)**plane** *npath* *npath* (\*)**zcut** *zcut* (\*)**xytol** *xytol* (\*)**all select** *ato4*

A plotfile is only produced if **\*plot** is given. If the plot plane is not a crystallographic mirror plane (as will normally be the case), then the gradient trajectories will, in general, leave the plot plane at some points. To force the plotted trajectories onto the plot plane, use the **\*plane** option. On the other hand, if plotting in a crystallographic mirror plane, then it is probably better not to use this option. *npath* is the number of downhill gradient trajectories - select an integer factor of 360. If **\*zcut** is selected, only those atoms within  $\pm$  *zcut* from the plot plane are included. If **\*xytol** is selected, only those atoms within  $\pm$  *xytol* from the plot plane edges are included. Either all atoms (**\*all**) are processed or only those selected atoms given (**\*select**).

## 6.6.7 SOURCE

Bader and Gatti [55] have shown that it is possible to view the electron density at any point **r** within a molecule to be determined by a source operating at all other points **r'**. The local source at position vector **r** from **r'**,  $LS(\mathbf{r}, \mathbf{r}')$ , is given by

$$LS(\mathbf{r}, \mathbf{r}') = -(4\pi |\mathbf{r} - \mathbf{r}'|)^{-1} \nabla^2 \rho(\mathbf{r}) \quad (\text{Eq 6-13})$$

The *local source* is calculated at a point, along a line, or in a 2D or 3D grid (for computation of the *integrated source* function see Section 6.8). The only properties for which the source may be calculated are the density  $\rho$  and the electrostatic potential  $\text{esp}$ . If another function is selected, the program halts. It is normal practise to use bond critical points as the reference points, since these are minimally biased, but any arbitrary point may be chosen.

**SOURCE retpoint** <*x,y,z*>

This instruction defines a reference point **r** in global Cartesian coordinates (Å) for the source calculation. Up to 100 reference points may be defined.

**SOURCE point** <x,y,z>

The local source is calculated at the point  $\mathbf{r}'$  <x,y,z>, given in global Cartesian coordinates (Å), for each of the reference points  $\mathbf{r}$ .

**SOURCE LINE** atom1, atom2 npts npts

The local source will be calculated at npts along the line defined by atom1 and atom2, for each of the reference points  $\mathbf{r}$ .

**SOURCE MAP atoms** atom1, atom2, atom3 npts npts **stepsize** step**SOURCE CUBE atoms** atom1, atom2, atom3 npts npts **stepsize** step

2D or 3D gridfile of the local source will be produced (xd\_source.grd & xd\_source.cube), with the definition of the plane given in the same way as the MAP instruction (see Section 6.1.16). Only one gridfile is produced with the **SOURCE MAP** or **SOURCE CUBE** options. If more than one source **SOURCE repoint** has been specified, then the source is calculated for the last repoint given.

## 6.7 Topological Analysis

### 6.7.1 CPSEARCH

**CPSEARCH bond** atom1 atom2 (**frac** frac) (**scan** delta)**CPSEARCH bond rmin** rmin **rmax** rmax**CPSEARCH ring** atom1 atom2 ...**CPSEARCH point** x y z**CPSEARCH line vec1** x1,y1,z1 **vec2** x2,y2,z2 **npts** npts**CPSEARCH shell** atom **rmin** rmin **rmax** rmax **nrad** nrad **nang** theta phi **cutoff** cutoff**CPSEARCH start** filename**CPSEARCH bubble** atom **rmin** rmin **rmax** rmax **curv** icurv **ncps** npoints

Topological analysis of the density is concerned with the scalar fields  $\rho$  and  $\nabla^2\rho$  but XDPROP allows the user to CP search *any* of the properties listed previously. In the case of the nuclear and electrostatic potentials, these have very similar topologies to  $\rho$  [46]. The usefulness of CP searching the **defden** is probably restricted to finding local (3,+3) minima or (3,-3) maxima, the latter (for example) would be bonded or non-bonded peaks in the **defden**.

The search for CPs employs a 3-dimensional Newton-Raphson technique, which requires both the gradient vector and the Hessian  $H$  (i.e. the  $3 \times 3$  matrix of partial second derivatives of  $f$  with respect to  $\{x,y,z\}$ ). Starting from some point, such as the midpoint between two nuclei, an improved estimate of a CP position  $\mathbf{r}_c$  is found from iterative application of

$$\mathbf{r}_c^{(n+1)} = \mathbf{r}_c^{(n)} - H^{-1} \cdot \nabla f \quad (\text{Eq. 6-13})$$

Each card beginning with **CPSEARCH** initiates a search of the property flagged with an asterisk on the **PROPERTY** card. Following **CPSEARCH** is a string which indicates how the searching is to be done. There are six modes of searching, which differ in how the starting coordinates are chosen, and how many times the CP search algorithm will be called.

- (i) **bond** can initiate a CP search between the two atoms whose labels follow, e.g.

```
CPSEARCH  BOND  C(1)  O(1)
```

This type of search starts from the midpoint of the two nuclei. The **frac** option can be used to change this.

```
CPSEARCH  BOND  C(1)  O(1)  FRAC  0.7
```

This starts the CP search at a fractional distance of 0.7 times the C-O bondlength, i.e. nearer to the oxygen. For bonds CPs which are tricky to find, a more thorough alternative is

```
CPSEARCH BOND C(1) O(1) SCAN 0.3
```

This carries out seven CP searches starting from (i) the midpoint of the nuclei  $\{x_o, y_o, z_o\}$  as before, then at  $\{x_o \pm \delta y_o, z_o\}$ ,  $\{x_o, y_o \pm \delta, z_o\}$  and  $\{x_o, y_o \pm \delta, z_o \pm \delta\}$ , where delta is the step (in Å) which appears after scan.

Alternatively if **bond** is followed by **rmin** instead of an atom label, then the program will search for all CP's between all pairs of nuclei with internuclear separations between *rmin* and *rmax* (in Å). So typically, to locate all CP's between bonded first-row atoms one might use:

```
CPSEARCH BOND RMIN 1.15 RMAX 1.6
```

- (ii) **ring** carries out a CP search starting at the centroid of the atomic coordinates of the atom labels which follow, so typically for a phenyl one could use something like:

```
CPSEARCH RING C(1) C(2) C(3) C(4) C(5) C(6)
```

At least two atom labels should be supplied, and no more than eight.

- (iii) **point** starts a search from the coordinates (in Å) which follow. These must be coordinates in the global Cartesian system, which is defined near to the top of the *xd\_pro.out* file. So the form of the command is:

```
CPSEARCH POINT 1.2 -4.3 9.3
```

- (iv) **line** starts a search along the line from the point defined by **vec1** to point defined by **vec2**. Both points are defined in the global Cartesian system. A total of *npts* points along this line will be used as starting points for the CP search, including the two defining points. So the form of the command is:

```
CPSEARCH LINE VEC1 1.2 -4.3 9.3 VEC2 2.2 -5.3 10.3 NPTS 15
```

- (v) **shell** is the most complex searching option, primarily intended for locating non-bonded charge concentrations in  $\nabla^2\rho$ ,  $\Delta\rho$ , or  $V(\mathbf{r})$ . An example would be:

```
CPSEARCH SHELL O(1) MIN .5 MAX .7 NRAD 3 NANG 11 11 CUT 20.
```

This searches in concentric spherical shells around O(1), with radii 0.5, 0.6 and 0.7 Å (since the number of shells *nrad*=3), over an  $11 \times 11$  angular grid of points in  $\theta$  and  $\phi$  (polar coordinates, referred to the global cartesian system). The minimum number of points in both  $\theta$  and  $\phi$  (**nang**) is two, and the minimum number of radial shells (*nrad*) is one. The algorithm operates in the following manner. Denoting the property being searched by  $f(\mathbf{r})$ ,  $|\nabla f(\mathbf{r})|$  is computed at each point  $\{r, \theta, \phi\}$  in the shell ( $3 \times 11 \times 11 = 363$  points, in the above example). If a point is found where  $|\nabla f(\mathbf{r})| < 20.0$ , then the Newton-Raphson CP search algorithm is initiated at this point, to see if a nearby CP can be located precisely. This routine often finds the same CP many times. Useful values of *cutoff* vary so much with the property to be searched, the radial distance from the nucleus, and the atomic number of the atom, that it is difficult to give guidelines on sensible values (currently there are no default values in the program). Subsequent releases of the program may be more helpful in this respect, but for the moment it is a matter for trial and error, and perseverance! The example given above has successfully found both carbonyl oxygen lone pair (3,-3) critical points in  $\nabla^2\rho$  in low-temperature data sets for formamide and acetamide.

- (vi) **start** reads in a data file generated by a previous run of the properties program, and carries out a sequence of CP searches using the CP positions given in that file as starting coordinates. This may be useful (i) if the refinement model has been altered,

but a set of CP positions have been computed for a previous model (ii) since it allows CP positions in one property to be used as starting points for searching another property. For example, the CP's in  $V(\mathbf{r})$ , which are very rapidly computed, may provide useful starting points for searching  $\rho$ . An example of the command would be:

```
CPSEARCH  START  XD_FORMAMIDE.CPS
```

where the last string is a filename.

- (vii) **bubble** allows a complete search of critical points around a given atom, particularly useful when searching critical points of the Laplacian. **NOTE** : unlike all other CPSEARCH options, **bubble** searches the function  $L(\mathbf{r}) \equiv -\nabla^2\rho$  when **property** is set to d2rho (*i.e.* critical points will have inverted signature compared with all other search options):

```
CPSEARCH BUBBLE C(1) rmin 0.3 rmax 0.5 curv -3 ncps 2
```

**curv** is the signature of the critical point to be searched. **ncps** is the expected number of CP's of this type around atom `ato(1)`. Setting **ncps** to zero allows the search to continue until finished. Enlarge *nstep* (in the heading of `xdprop`) if a more thorough search is needed. A file `xd_bubble.spf` is created with all CP's found, suitable for rendering in PLATON [47].

**Note:** in this version of XDPROP, only the unique set of critical points is stored (previous versions allowed multiple finds).

### 6.7.2 BPATH (Bond Path analysis)

#### BPATH *atom1 atom2* **algorithm** (2|6) (**profile**)

The bond path (BP) is the line of maximum charge density joining two nuclei. It passes through the CP, and the technique for computing the BP trajectory is to carry out separate line integrations of  $\nabla\rho$  from the CP towards the two nuclei. The initial direction for this integration is determined by the CP eigenvector with the positive eigenvalue, approximately parallel to the internuclear vector. An analysis of the BP is mainly useful for detecting strain in bonds [34], since in such bonds the BP is significantly longer than the internuclear vector. Other parameters of interest are the take-off angles of the BP at the CP and the nuclei. These are angles between the BP and the internuclear vector.

There are two BP-determining algorithms in XDPROP, with different levels of accuracy. The simplest uses an Euler second-order numerical integration technique, requiring only  $\nabla\rho$  at a point to extrapolate to the next step. The second is a sixth-order algorithm, which uses a number of prior points to extrapolate to the next step, and ought to be rather more accurate. BP files may be produced, which contain the coordinates of BPs at regular intervals along the path, so that they may be plotted using the graphics software supplied in XD.

The command format is simple:

```
BPATH  C(1) O(1)  algorithm 2
```

computes the **BPATH** between C(1) and O(1) using the second-order algorithm, simultaneously producing a BP file for plotting. The more accurate sixth-order algorithm is selected by replacing the "2" with a "6" in the above example. An experimental modification of the sixth-order algorithm is also available by replacing "6" with "-6".

If the option **profile** is added to the command, a complete Bond Path Analysis [53] ( $\rho(\mathbf{r})$ ;  $\nabla^2\rho(\mathbf{r})$  and  $\epsilon$ ) is performed and reported in the file `xd_profile.dat`.

```
BPATH  ato(1) ato(2) algorithm 2 profile
```



Here is an example of `xd_profile.dat`

```
#
# bond path 1 between atoms FE and C(1)
#
#      r (A)      ellip      rho (e/A^3)      Laplacian (e/A^5)
#      0.0000      0.1249      1.0201      7.0729
#     -0.0060      0.1275      1.0205      7.5554
#     -0.0110      0.1296      1.0212      7.9684
#     .....      .....      .....      .....
```

The origin is fixed at the *bcp*, two "legs" are printed (one for each atom). If more than one **BPATH** command is requested, then the following bond path analyses are simply appended. If more points along the profile are required, then decrease the value of the *pstep* parameter from its default value of 0.005 (see **SELECT ds** above).

If the option **check** is flagged, the `xdprop` output file `xd_pro.out` contains full analysis of each point along the bond path integration (density, Laplacian, Hessian, diagonalisation of the Hessian matrix, direction of the eigenvectors, etc.).

### 6.7.3 MOLGR (Automatic Molecular Graph Analysis)

**MOLGR \*auto level nlevel rmin rmin rmax rmax algrithm (2|6) (profile)**

This directive provides for an automatic search for the full molecular graph. It uses the **CPSEARCH** and **BPATH** routines described above, though their output has been simplified when they are run under the **MOLGR** directive. Note that this option uses ALL the atoms in the current atom list (including those added/removed through **APPLY/GROUP/OMIT/CRYSTAL** commands). So, if a molecule sits on a symmetry element, it will be necessary to explicitly include the required **APPLY/etc** commands.

If **auto** is flagged (*i.e.* starred, the default), **MOLGR** instigates a fully automatic procedure to determine the atomic graph, otherwise the procedure relies on immediately previous **CPSEARCH** instructions (*e.g.* **CPSEARCH start**) which have located all the critical points.

The **level** setting *nlevel* determines how comprehensive the automatic critical point search is. There are currently three allowed *nlevel* settings 1, 2 or 3. For *nlevel* = 1, the automatic procedure firstly uses the mid-points of all pairs of atoms which fulfill the **rmin/rmax** conditions as starting points for the Newton-Raphson search. This will generally find all the bond critical points, and if **rmax** is set sufficiently high, a number of ring critical points may also be obtained. If the resultant set of critical points satisfies the Poincaré-Hopf relationship, *i.e.*  $\text{nat-nbcp} + \text{nrcp} - \text{nccp} = 1$ , the procedure then immediately moves onto the tracing of the bond paths. For this, the **algrithm (2|6) (profile)** directives are used exactly as for the **BPATH** command given above. If on the other hand the Poincaré-Hopf relationship is NOT satisfied, then a second cp search is instigated, which uses all triads of atoms satisfying the **rmin/rmax** conditions to find any missing ring and/or cage critical points. As soon as the Poincaré-Hopf relationship is satisfied, this procedure exits and moves onto the bond path tracing.

For *nlevel* = 2, the procedure is similar, except that (i) the **rmin/rmax** conditions are not used, *i.e.* ALL pairs/triads are used for the Newton-Raphson search, and (ii) the Poincaré-Hopf relationship is checked but never used to exit the routine, *i.e.* both sets of cp searches are carried to completion. This is naturally a more time-consuming process, but is usually worthwhile, in order to be certain that the full graph has been obtained. It is important to be aware that the Poincaré-Hopf relationship is a *necessary* but not *sufficient* condition for the completeness of the molecular graph, so the fact the relationship is satisfied is no guarantee that the molecular graph is actually complete.

The *nlevel* = 3 setting is experimental, and uses a comprehensive grid search for critical points. It is very compute-intensive, untested, and is only intended for desperate situations!

## 6.8 Atomic Basin Integration

Atomic properties are integrated in the atomic basins. Most of the properties integrated by **TOPXD** (see Section 10.9) are also provided with **TOPINT**, and in addition, the integrated source function may be computed. **TOPINT** works either in the *crystal* environment, or in the *isolated molecule (gas-phase)* environment (similar to many quantum calculations), depending on whether the **CRYSTAL** directive is active. This is in contrast to the **TOPXD** program (Section 10), which designed to work fairly exclusively in the *crystal* environment.

The relevant instructions in the `xd.mas` file are :

**EXPORT ....**

**ORIGIN \*mass (\*)charge (\*)geom (\*)ucell**

**QUADINT iqt 2 Nrad Nrad Nang Nang (\*)Becke (\*)Stock**

**ODESOLVE (\*)rk (\*)bs eps eps stepi stepi**

**CRYSTAL alim xmin xmax blim ymin ymax clim zmin zmax**

**SOURCE refpoint <x,y,z>**

**TOPINT step step accur accur rmax rmax**

**TOPINT spheres ato(1) r1 ato(2) r2 ....**

**TOPINT atoms (\*)all \*unique (\*)select ato(1) ato(2) ...**

The only parameters used from the **QUADINT** instruction are **Nrad** and **Nang**, the number of radial and angular quadrature points. The value of **Nrad** can be set large, as it does not greatly affect the CPU time. Larger values of **Nang** should result in better integration, but are more time consuming. Gauss-Legendre quadrature is used for the radial integration, and Lebedev quadrature [15] for the angular integration. Any value may be set for **Nang**, the program automatically assigns the closest allowed Lebedev quadrature. Allowed values of **Nang** range from 6 - 5810.

For the parameters associated with the **ODESOLVE** directive, see section 6.4.3. If the Runge-Kutta (**\*rk**) algorithm gives step underflow errors, then switch to the Bulirsch-Stoer (**\*bs**) method.

If the **CRYSTAL** directive is absent, the integration is carried out over the cluster of atoms as defined normally in XDPROP (see Section 6.3).

**SOURCE refpoint <x,y,z>**

Reference point **r** for source calculation, <x,y,z> in global Cartesian coordinates

The presence of **SOURCE refpoint** directives activates the calculation of the integrated form of the source function for each atomic basin specified in the **TOPINT atoms** directive described below, at each of the given reference point(s) **r**. Up to 100 reference points may be specified. The reference points would normally be bond critical points (as the most unbiased reference points) but any point may be specified.

**IMPORTANT:** The only properties for which the source may be calculated are the density  $\rho$  and the electrostatic potential  $\text{esp}$ . If another function is selected, the program ignores any **SOURCE refpoint** instructions. All the **SOURCE refpoint** instruction(s) MUST precede any **TOPINT** instructions in the `xd.mas` file.

The local source ( $LS$ ) contribution at position vector **r** from **r'** is given above by Eq 6-13. By integrating over the regions of space bounded by the zero-flux surfaces, the density may be equated to a sum of atomic contributions  $S(\mathbf{r}, \Omega)$ .

$$\rho(\mathbf{r}) = \sum_{\Omega} \int_{\Omega} LS(\mathbf{r}, \mathbf{r}') d\mathbf{r}' \equiv \sum_{\Omega} S(\mathbf{r}, \Omega) \quad (\text{Eq 6-14})$$

The integrated form of the source function (SF)  $S(\mathbf{r}, \Omega)$  provides a measure of the relative importance each atom has in determining the density at a specific point [55]. This property has great power in illustrating the non-local properties of the electron density.

Note that Eq 6-14 is only useful for decomposing  $\rho(\mathbf{r})$  into contributions from the integrated source function of atomic basins in the case of a *closed system* where all zero flux surfaces terminate at infinity (*e.g* in an isolated molecule). For a molecule in a crystal, the density at point  $\mathbf{r}$  within atomic basin  $\Omega$ , is alternatively given [55] by Eq (6-15)

$$\rho(\mathbf{r}) = -(1/4\pi) \left\{ \int_{\Omega} \frac{\nabla^2 \rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r}' + \oint_{\partial\Omega} dS(\mathbf{r}_s) \cdot \boldsymbol{\varepsilon}(\mathbf{r} - \mathbf{r}_s) \right\} \quad (\text{Eq 6-15})$$

The first integration term represents the basin average of the potential at  $\mathbf{r}$  exerted by the Laplacian of the density, while the second integration term represents the flux through the surface boundary of  $\Omega$  of the electric field density,  $\boldsymbol{\varepsilon}(\mathbf{r} - \mathbf{r}_s)$  at  $\mathbf{r}$  due to the density, and corresponds to the source contributions from the basins of all other atoms.

**IMPORTANT:** The current version of **TOPINT** does not calculate this surface integral, so the source function for a *crystal* environment cannot be properly calculated.

### 6.8.1 TOPINT Instructions

**TOPINT** **step** *step* **accur** *accur* **rmax** *rmax* **nmax** *nmax*

**step** *step* [0.01]

The step distance along the integration ray, in Angstroms.

**accur** *accur* [0.001]

The required accuracy in the final surface determination in Angstroms. Surface determination is done via the bisection method. In order to use step-by-step method, set **step** = **accur**. In principle, the smaller the values of **step** and **accur** the better, but this becomes much more time consuming. It does not make sense to set **step** smaller than **accur**.

**rmax** *rmax* [8.0]

The maximum allowed length of the integration ray, in Angstroms.

**nmax** *nmax* [10]

The maximum allowed number of atoms in the cluster around the integrated atom. The default value is probably suitable for most organic compounds, but will probably need to be increased for inorganic and organometallic compounds.

**TOPINT spheres** *ato(1) r1 ato(2) r2 ....*

**spheres**

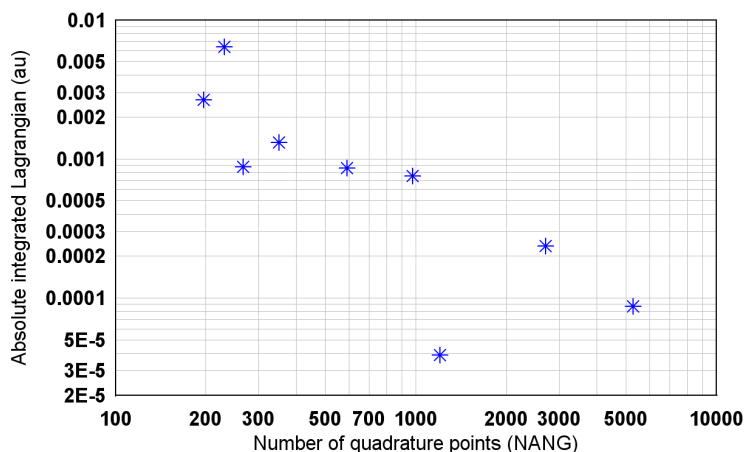
The  $\beta$ -sphere radii for individual atoms may be given explicitly. These would normally be slightly smaller than the distance to the closest bond critical point. If no **spheres** instructions are given, then the  $\beta$ -sphere radii for all atoms are calculated automatically, and are printed in the output file `xd_pro.out`. These radii can then be used in subsequent **TOPINT** jobs.

**TOPINT atoms** (\*)**all** \***unique** (\*)**select** *ato(1) ato(2) ...*

**atoms**

Specifies the atoms to be integrated. **\*all** indicates that all atoms in list are to be integrated, **\*unique** specifies just the asymmetric unit, while **\*select** is followed by a list of specific atoms to integrate.

No errors for the integrated properties are obtained, but the accuracy of the integration may be gauged by the value of the integrated Lagrangian, (property L) which should ideally have a value of zero. The smaller the value the better, and it should ideally be below  $1.0 \times 10^{-4}$  au. Increasing the value of **Nang** in the **QUADINT** instruction will generally lead to a better integration, at the expense of increased CPU time, but the response is by no means linear and increasing **Nang** may even cause the Lagrangian to increase ! see Fig 6-3.



**Fig 6-4 Typical plot of the integrated Lagrangian function versus number of angular quadrature points (Nang) for integration of the oxygen atom in water dimer (log-log scaling) using TOPINT.**

## Bibliography

1. R.F. Stewart and M.A. Spackman, *VALRAY users manual*, Department of Chemistry, Carnegie-Mellon University, Pittsburgh, PA, U.S.A.
2. N. K. Hansen and P. Coppens, *Acta Cryst.* **A34**, 909-921 (1978).
3. (a) P. Coppens *X-ray charge densities and chemical bonding*, Oxford University Press, 1997. (b) V. G. Tsirelson and R. P. Ozerov *Electron density and bonding in crystals*, Institute of Physics Publishing, Bristol, 1996. (c) T. Koritsanszky and P. Coppens, *Chem. Rev.* **101**, 1583 (2001). (d) P. Macchi and A. Sironi, *Coord. Chem. Rev.* **238-239**, 383 (2003). (e) C. Gatti, *Z Kristallogr.* **220**, 399 (2005).
4. E. Prince, in *Mathematical Techniques In Crystallography And Materials Science*, Springer-Verlag: New York, 1982.
5. D.W.J. Cruickshank and A.P. Robertson, *Acta Cryst.* **6**, 698 (1953).
6. Z.W. Su and P. Coppens, *Acta Cryst.* **A48**, 188 (1992).
7. G. te Velde, F. M. Bickelhaupt, E. J. Baerends, C. Fonseca Guerra, S. J. A. van Gisbergen, J. G. Snijders, T. Ziegler, *J. Comp. Chem.* **22**, 931 (2001).
8. BAND - A Fortran program for Band Structure Calculations. B te Velde (1990)
9. Modern Density Functional Theory, N. C. Handy .
10. A. D. Becke, *J. Chem. Phys.* **88**, 1053 (1988).
11. J. M. Perez-Jorda, A. San-Fabian, F. Moscardo, *Comp. Phys. Comm.* **70**, 271 (1992).
12. O. Treutler, R. Ahlrichs, *J. Chem. Phys.* **102**, 346 (1995).
13. M. Abramowitz, I. E. Stegun "Handbook of Mathematical Functions" (1970).
14. M. E. Mura, P. J. Knowles, *J. Chem. Phys.* **104**, 9848, 1996) ; P. M. W. Gill, S.-H. Chien, *J. Comp. Chem.* **24**, 732 (2003).
15. V. I. Lebedev, D. N. Laikov, *Doklady Math.* **59**, 477 (1999).
16. F. L. Hirshfeld, *Theor. Chim. Acta* **44**, 129 (1977).
17. M.A. Spackman, *Chem. Rev.* **92**, 1769 (1992).
18. P. Coppens and N.K. Hansen, *Israel J. Chem.* **16**, 163 (1977).
19. P. Coppens, G. Moss and N.K. Hansen, in *Computing In Crystallography*, Eds. R. Diamond, S. Ramaseshan, K. Venkatesan, Indian Academy of Sciences, Bangalore, 1980.
20. F.L. Hirshfeld, *J. Mol. Struct.* **130**, 125 (1985).
21. A. J. Stone, *The Theory of Intermolecular Forces*; International Series of Monographs on Chemistry 32; Oxford University Press; Oxford, 1996.
22. A. J. Stone, Program ORIENT, <http://www-stone.ch.cam.ac.uk/programs.html>
23. Z. Kisiel, PROSPE - Programs for ROtational SPEctroscopy, <http://info.ifpan.edu.pl/~kisiel/prospe.htm>.
24. M. A. Spackman *J. Chem. Phys.* **85**, 6579 (1986).
25. D. E. Williams, S. R. Cox *Acta Cryst* **B40**, 404 (1984).
26. J. E. Lennard-Jones, *Proc. Phys Soc* **43**, 461 (1931).
27. A. Volkov, C. Nygren, M. Messerschmidt, X. Li, P. Coppens, in preparation (2006).
28. (a) A. Volkov, P. Coppens *J. Comp. Chem.* **25**, 921 (2004). (b) A. Volkov, Koritsanszky and P. Coppens, *Chem. Phys. Lett.* **391**, 170 (2004)
29. A. Volkov, X. Li, T. Koritsanszky, P. Coppens *J. Phys. Chem. A* **108**, 4283 (2004).

30. (a) M. A. Spackman *J. Chem. Phys.* **85**, 6579 (1986), (b) *ibid.*, 6587 (c) M. A. Spackman *J. Chem. Phys.* **91**, 3179 (1987),
31. A. D. Buckingham *Intermolecular Interactions : from Diatomics to Biopolymers* ed. B. Pullman, pp 1.67, New York, Wiley, Chichester; A. D. Buckingham *Adv. Chem. Phys.* **12**, 107 (1967).
32. A. Holladay, P. Leung and P. Coppens, *Acta Cryst* **A39**, 377-387 (1983).
33. K.E. Laidig, *J. Phys. Chem.* **97**, 12760 (1993).
34. J.D. Jackson, *Classical Electrodynamics*, J. Wiley & Sons: New York, p98, 1963.
35. G. Moss and D. Feil, *Acta Cryst.* **A37**, 414 (1981).
36. R. Destro, R. Bianchi and G. Morosi, *J. Phys. Chem.* **93**, 4447 (1989).
37. W.T. Klooster, S. Swaminathan, R. Nanni and B.M. Craven, *Acta Cryst.* **B48**, 217 (1992).
38. N. Ghermani, N. Bouhaida, C. Lecomte, A. Papet and A. Marsura, *J. Phys. Chem.* **98**, 6287 (1994).
39. R.K. Pathak and S.R. Gadre, *J. Chem. Phys.* **93**, 1770 (1990).
40. A. Kirzhnits, *Sov. Phys JETP*, **5**, 65 (1957).
- 41a. V. Tsirelson and A. Stash, *Chem. Phys. Lett.*, **351**, 142 (2002).
- 41b. V. Tsirelson and A. Stash, *Acta Cryst* **B58**, 780 (2002)
42. D. Jayatilaka, D. Grimwood *Acta Cryst.* **A60**, 111 (2004).
43. E. N. Lassette, *J. Chem. Phys.*, **83**, 1709, (1985).
44. G. Hunter, *Int. J. Quantum Chem.*, **29**, 197 (1986).
45. M. Kohout, *Int. J. Quantum Chem.*, **83**, 324 (2001).
46. R. F. W. Bader, *Atoms in Molecules: A Quantum Theory*, Oxford University Press, 1990.
47. PLATON, A Multipurpose Crystallographic Tool, Utrecht University, Utrecht, The Netherlands, A. L. Spek, 2003.
48. D.E. Williams and S.R. Cox, *J. Comput. Chem.* **2**, 304 (1981).
49. D.E. Williams, *J. Am. Chem. Soc.* **105**, 4143 (1983).
50. C.M. Breneman and K.B. Wiberg, *J. Comp. Chem.* **11**, 361 (1990).
51. Z.W. Su, *J. Comp. Chem.* **14**, 1036 (1993).
52. R.F.W. Bader and H. Essén, *J. Chem. Phys.* **80**, 1943 (1984).
53. M. Tafipolsky, W. Scherer, K. Ofele, G. Artus, B. Pedersen, W. Herrmann, G. S. McGrady *J. Am. Chem. Soc.* **124**, 5865 (2002).
54. MOLEKEL is a free graphics program available from <http://www.cscs.ch/molekel/>
55. (a) R. W. F. Bader and C Gatti *Chem. Phys Lett* **287**, 233 (1998). C. Gatti, F. Cargnoni, L. Bertini *J. Comput Chem* **24**, 422 (2003)
56. A. Volkov, H. F. King, P. Coppens and L. J. Farrugia *Acta Cryst* **A62**, 400-408 (2006).
57. M. A. Spackman and D. Jayatilaka *CrystEngComm* **11**, 19-32 (2009).
58. E. R. Johnson, S. Keinan, P. Mori-Sánchez, J. Contreras-García, A. J. Cohen and W. Yang *J. Am. Chem. Soc.* **132**, 6498 (2010).
59. G. Saleh, L. Lo Presti, C. Gatti and D. Ceresoli *J. Appl. Cryst.* **46**, 1513 (2013).
60. P. de Silva and C. Corminboeuf *J. Chem Theory Comput.* **10**, 3745 (2014).
61. P. de Silva, J. Korchowiec and T. A. Wesolowski *ChemPhysChem* **13**, 3462 (2012).

