

EDMA - A computer program for analysis of
electron-density maps in arbitrary dimension

User Manual

Lukáš Palatinus
Sander van Smaalen

version: 06/11/2006

Contents

1	Introduction	3
2	Theoretical foundations	4
2.1	Discretized electron density in 3D and nD	4
2.2	Definition of position of atoms	4
2.3	Charge partitioning and atomic basins	5
2.4	Critical points	5
2.5	Interpolation	6
3	Technical details	8
3.1	Programming language and system requirements	8
3.2	Execution	8
3.3	Standard run of the program	8
4	Specification of the input	10
4.1	Types of input	10
4.2	Format of the ASCII input file	10
4.3	Specification of keywords	10
4.3.1	name: addborder	11
4.3.2	name: edmaatoms - endedmaatoms	11
4.3.3	name: axisorder	12
4.3.4	name: basins	12
4.3.5	name: cell	12
4.3.6	name: centerofcharge	12
4.3.7	name: centers - endcenters	13
4.3.8	name: chlimit	13
4.3.9	name: chlimlist	13
4.3.10	name: composition	13
4.3.11	name: cplist - endeplist	14
4.3.12	name: criticalpoints	14
4.3.13	name: export	14
4.3.14	name: fullcell	14
4.3.15	name: inputfile	15
4.3.16	name: maxima	15
4.3.17	name: numberofatoms	15
4.3.18	name: outputbase	15
4.3.19	name: plimit	15
4.3.20	name: position	16
4.3.21	name: projection	16
4.3.22	name: qvectors - endqvectors	16
4.3.23	name: range	16
4.3.24	name: scale	17

4.3.25	name: symmetry - endsymmetry	17
4.3.26	name: tlist - endtlist	17
4.3.27	name: tmap	17
4.3.28	name: tolerance	18
4.3.29	name: writem40	18
4.4	Examples of input files	18
4.4.1	Example 1.	18
4.4.2	Example 2.	18
4.4.3	Example 3.	19
4.4.4	Example 4.	19
4.4.5	Example 5.	20
5	Description of the output	21
5.1	File outputbase.coo	21
5.2	Files outputbase.tmp and outputbase.tmc	23
5.3	t-maps	24
5.4	Atomic basins	24

Chapter 1

Introduction

The Maximum Entropy Method (MEM) is nowadays well established as a powerful tool for a model-free image reconstruction. In crystallography it has found applications in the reconstruction of the electron density in the unit cell (von der Linden et al., 1998). One of its promising applications is the reconstruction of the electron density of the incommensurately modulated structures in the higher-dimensional unit cell. This can be used for a model-free determination of the shapes of the modulation functions, a result which is not achievable with the standard refinement methods.

However, the output from the MEM is a discrete n -dimensional array of values of the electron density representing a sampling of the true continuous electron density. Such a type of output does not allow a straightforward quantitative interpretation and extraction of all the relevant information, like the position of the atoms, charge partitioning between the atoms and ultimately the shape of the modulation functions. It is precisely the task of the program EDMA to provide the necessary software tools for such an analysis and extraction. The program can extract an exact position of the atomic maxima from any discrete electron density map, the charge of individual atoms and the center of the charge, topological properties of the electron density, all this as a function of the superspace coordinates in case of higher-dimensional densities.

EDMA is optimized for work with the superspace densities, but since the 3D densities are but a special case of the general nD density, EDMA is fully capable of the analysis of ordinary 3D-periodic structures, too.

Although EDMA is primarily designed for analysis of the electron density maps calculated by the MEM, it can be successfully used to analyze any electron density, including those created by standard Fourier transform of phased structure factors or calculated by *ab initio* phasing methods such as charge flipping.

Chapter 2

Theoretical foundations

2.1 Discretized electron density in 3D and nD

The output of the MEM applied to the reconstruction of the electron-density distribution is an array of values of the electron density (electron-density map or simply map). The points at which the density is calculated are distributed on a regular grid in the unit cell (pixel grid). The number of pixels along each direction can be theoretically arbitrary, but it is restricted mainly by the computer speed. The size of the pixels is nowadays usually around 0.05-0.1Å and it is likely to decrease with the increasing computational power becoming available.

Three-dimensional (3D) periodic structures can be uniquely characterized by a 3D density map of one unit cell. Atoms can be easily identified as local maxima of the electron density. Non-periodic structures, like incommensurately modulated structures, composite structures or quasicrystals, can be described by (3+d) dimensional periodic electron density (superspace density), where d stands for the number of additional dimensions perpendicular to the physical space. In such a superspace electron density the atoms form wavy strings ($d = 1$), distorted planes ($d = 2$) or hyperplanes ($d > 2$). Each 3-D section of the superspace density perpendicular to the additional dimensions represents an arrangement of the atoms in the physical structure. These sections are 3D, but they are not periodic. Different sections differ only by a shift of the origin. All possible arrangements of the atoms can be obtained by considering sections of only one superspace unit cell. These sections (t-sections, t-maps) can be used to extract the atomic position and topological properties at given t. The modulation function can be reconstructed from the positions of the atom in different t-maps. For more information on the problem of aperiodic crystal structures see specialized literature, e.g. van Smaalen (1995).

2.2 Definition of position of atoms

The atomic position in a 3D map (either periodic or a t-map) can be defined in two ways. First, it can be defined as position of a local maximum of the electron density. This definition is straightforward and it is correct for atoms, which do not exhibit anharmonic thermal displacements. Second, the atomic position can be defined as the center of the charge of the electron density belonging to the supposed atomic position. This is equivalent to a center of atomic mass, if we assume the electrons follow the nuclear movement exactly, the stationary atom is spherically symmetric and we neglect the deformation due to the bonding, all these assumptions being reasonable. This second definition is closer to the definition used in standard refinement techniques.

EDMA can calculate both types of atomic positions. By comparing them an indication of the anharmonic thermal movement can be found. As a technical detail, not all pixels

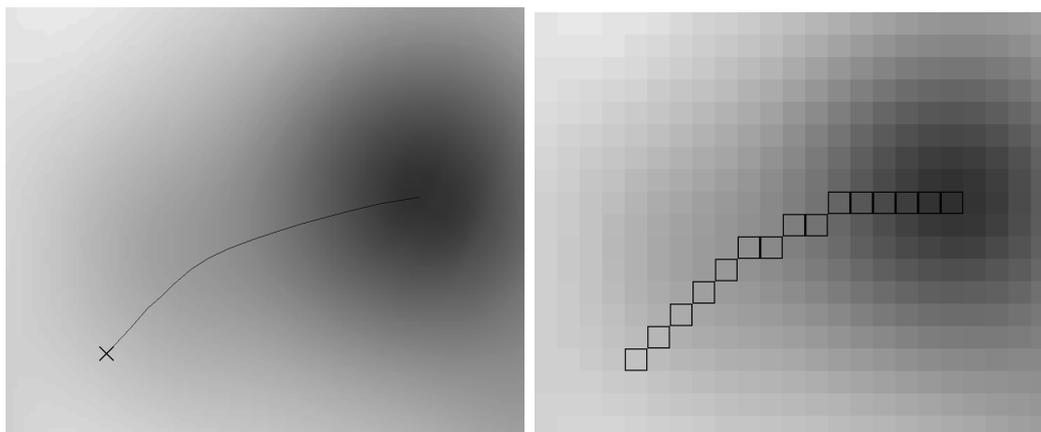


Figure 2.1: The assignment of a point to a local maximum a) in a continuous density according to the theory and b) in a discrete density by choosing the neighboring pixel of the highest density until the maximum is reached.

belonging to the atomic basin of particular atom (see section 2.3) are included in the calculation of the center of charge. Doing so would include the low-density pixels that may contain a lot of noise and thus may introduce errors in the result. Therefore, only pixels with densities above some (user-definable) fraction of the maximum density are taken into account for the calculation of the center of charge.

2.3 Charge partitioning and atomic basins

One of the characteristics of an atom is its total charge. The concept of atomic charge is quite complicated and there are several approaches to it (Hirshfeld (1977), Bader (1990), Coppens (1997)(chapter 3) and references therein). Probably the most widely used is the theory "Atoms in molecules" (Bader, 1990). In this theory, space is partitioned into atomic basins, based on the gradient of the electron density. The atomic basins are non-overlapping and space-filling. The charge of the atom is defined as an integral of the charge over the atomic basin.

It is this space- and charge-partitioning system that is used in EDMA to calculate the charge of the atom (and to calculate the center of charge). The method has been adapted to discretized electron density and simplified to allow reasonably fast calculation. (Fig. 2.3) The charge of each pixel is assigned to the corresponding local maximum fully. This is not a problem except for the pixels on the border of two atomic basins. The border would probably divide the pixel in two parts, assigning each part to different maximum. Thus, small inaccuracies in the charges can be expected. In particular, two symmetrically equivalent atoms, which share a common border, will have slightly different charges, because the pixels on their border are assigned arbitrarily to one of them. This problem can be overcome by prior resampling of the electron density to a better resolution. However, this will naturally increase the computational time. In practice, the inaccuracies introduced by the discrete nature of the MEM electron density are usually much lower than the inaccuracy resulting from other sources.

2.4 Critical points

EDMA searches the electron density for the critical points and determines their properties. Critical points of the electron density are points with zero density gradient:

$$\nabla\rho(\vec{r}) = 0 \quad (2.1)$$

There are four types of the critical points based on the signs of the principal curvatures at the point. These include:

- type (3,-3): Local maximum of the density. All three principal curvatures are negative.
- type (3,-1): Bond critical point. Two of the principal curvatures are negative and one is positive.
- type (3,+1): Ring critical point - occurs for example in the center of the benzene cycle. Two of the principal curvatures are positive and one is negative.
- type (3,+3): Pit critical point - local minimum of the electron density. All three principal curvatures are positive.

The properties determined at each critical points include the electron density $\rho(\vec{r})$, the principal curvatures $\lambda_i, i = 1, 3$, the eigenvectors of the Hessian matrix and the laplacian of the electron density at the critical point. For the physical meaning of the individual properties see (Bader, 1990).

2.5 Interpolation

The resolution of the grid is finite (about 0.05-0.10 Å) and therefore an interpolation method must be used to obtain the density and its derivatives at any point of the unit cell. In addition to that, the t-sections of the superspace electron density have general orientation with respect to the grid and their accurate extraction also demands interpolation of the density values between grid points.

The basis of the interpolation method used in EDMA is a cubic spline interpolation in one dimension (Press et al., 1996). The method has been generalized to arbitrary dimension using the philosophy of bicubic spline interpolation (Press et al., 1996). The principle of the method is illustrated in Fig. 2.2. The interpolation of a D-dimensional electron density runs in D cycles. In every cycle, one-dimensional cubic spline interpolation is used to calculate (k-1)D section of the kD density. This section contains the point, in which the density is to be determined. In the next cycle the newly calculated (k-1)D density is used as input and the dimension is further reduced. Ultimately, this leads to the interpolated density of the point in question. The result of the interpolation is independent of the order of axes, in which the density is interpolated.

The interpolated value by the cubic-spline method is in principle dependent on all values of the grid. In practice, the dependence strongly decreases with the distance from the central point. Therefore, In EDMA, every point is calculated from the pixels lying in some (user-definable) neighborhood of that pixel. This simplification does not introduce any detectable errors and saves computational time.

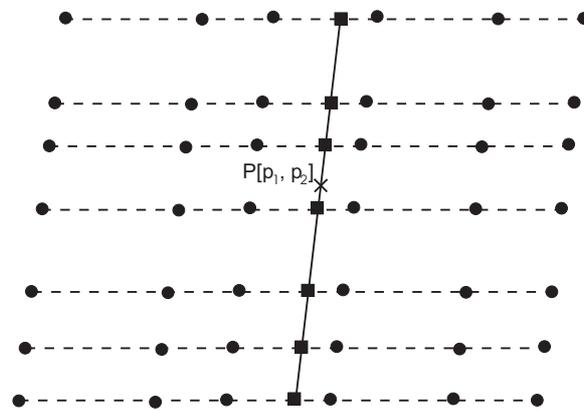


Figure 2.2: Interpolation process in two dimensions. In first step, the rows of points x_{ij}, y_{ij} (circles) are interpolated along the dashed lines to obtain points of the vector \vec{y}_{p_1} (squares). This column is then interpolated along the full line to obtain the density at the point P (cross).

Chapter 3

Technical details

3.1 Programming language and system requirements

EDMA is written in the programming language Fortran 90. It has been compiled and tested on two computers:

- Compaq AlphaStation ES40 with 500MHz 64-bit Alpha EV6 RISC processor and with Compaq Fortran Compiler V5.5-1877-48BBF
- Silicon Graphics Fuel with 500MHz IP35 MIPS R14000 processor and with MIPSPro Fortran compiler V7.4

The program obeys Fortran 90 standards and should be therefore compilable with any F90 compiler.

The program does not have any special system requirements. It does not use a graphical interface and the input can be edited with any plain text editor such as vi, nedit or emacs. However, it should be noted that the requirements for the system memory are quite high. The program stores the whole electron density and some smaller arrays in RAM to achieve reasonable speed of computation. Since the electron density maps can be very large (up to the order of several GB in case of high-resolution 4-D maps), the memory requirements are correspondingly high. However, these requirements are not higher than those of the program BAYMEM. Thus, any computer used for computation of the electron density map with BAYMEM can be used for analysis of that map with EDMA.

3.2 Execution

The program is executed with command

```
EDMA input_file
```

where `input_file` is the full name of the ASCII file containing input parameters (see Sections 4.2, 4.3 and 4.4). There are no other command-line options.

3.3 Standard run of the program

This is an example of a standard run of the program, if no error has been encountered in the input and during the execution:

```
Reading the density map Example.m81
t= 0.000
Extracting the electron density in real dimensions:
Integrating charge of the atoms:
t= 0.040
```

```
Extracting the electron density in real dimensions:  
Integrating charge of the atoms:  
67%
```

```
.  
.  
.
```

The last line shows always the progress of the current task in percent. The sentence "Integrating charge of the atoms" appears only if setting `centerof-charge yes` is present in the input file (Section 4.3.6).

The output goes standardly to the standard output. Should EDMA run after closing of the terminal it was launched from, the output must be redirected to some temporary file, for example:

```
EDMA input_file > edma.tmp
```

Chapter 4

Specification of the input

4.1 Types of input

EDMA requires two input files. The first file contains the values of the electron density. EDMA supports electron density files in the .m81 format of JANA2000 (Petříček & Dušek, 2000)¹ and the xplor format, that can be read for example by the crystallographic modeling and visualization package Chimera (<http://www.cgl.ucsf.edu/chimera/>). The density must be defined in points $x_i = 0, 1/n_i, 2/n_i, \dots, (n_i - 1)/n_i$ where n is number of pixel in each direction i . This is the standard produced by the programs BAYMEM and SUPERFLIP. Other formats can be added upon request.

The second file is an ASCII file containing specifications on the program parameters. In following sections, the expression “input file” means always the ASCII input file. The file with electron density is referred to as a “input density file” or “input density”.

4.2 Format of the ASCII input file

The input file is a free-format file based on keywords. Each keyword represents a specific parameter of the analysis and must be given a value.

Multiple spaces anywhere in the file are handled as a single space. The lines containing the character '#' in the first column are treated as comments and not interpreted. Blank lines anywhere in the input file are ignored. The length of the interpreted part of the line is 132 characters, every text exceeding this length is ignored.

4.3 Specification of keywords

There are two basic types of keywords. The first type is followed by one or more values on the same line:

```
keyword value1 [value2 value3...]
```

The second type has the form:

```
begin_keyword  
line 1  
line 2  
...  
end_keyword
```

¹This crystallographic package is free and can be downloaded at <http://www-xray.fzu.cz/jana/Jana2000/jana.html>

Each line may contain one or more values.

The name of the keyword of the first type is a single word without spaces. The name of the keyword of the second type is a pair of initial and final word (separated by a hyphen in the following text).

- Each value can be a constant of type real, integer or character. The type of the parameters and their allowed values are specified below. Alternative values are separated by slashes.
- The keywords are either compulsory or optional. The compulsory keywords must be specified for the analysis to proceed. The optional keywords can be omitted. If the optional keyword is omitted, the default value is used. The compulsory keywords are indicated by “compulsory keyword - no default” in the item “default”.
- description: Describes the function of the keyword, its influence on the output and relations to other keywords.

4.3.1 name: **addborder**

- value: positive real number
- default: 0
- description: The t-maps are non-periodic in one or more direction. However, some atomic basins span beyond the borders of the current basic-structure unit cell. The density of these parts of the atomic basins must be calculated explicitly in the aperiodic directions, if any characteristics depending on the atomic basins (see keywords **basin**, **centerofcharge**) should be computed. The keyword **addborder** controls, how large portion of the neighboring unit cell should be calculated for each t. The parameter gives the fraction of the neighboring unit cells to be added to the actual unit cell. For example, **addborder 0.4** will cause EDMA to calculate t-maps in interval -0.4 to 1.4 with respect to the actual unit cell in each aperiodic direction. It is user’s responsibility to use sufficiently large value of **addborder**. If the number is too small, not all charge of the atom might be accounted for and consequently the extracted properties of the atoms at the border of the unit cell can be biased. On the other hand, the computational time increases approximately linearly with the value of **addborder** ($t = t_{\text{addborder}=0}(1 + 2 * \text{addborder})$) and thus redundantly large values should be also avoided. The proper value depends on the particular crystal structure and can be determined by inspecting maps of atomic basins (see keyword **basins** and Section 5.4).

4.3.2 name: **edmaatoms - endedmaatoms**

- value: each line contains a name of an atom (maximum 8 characters) and three fractional basic-structure coordinates of that atom.
- default: compulsory keyword in combination with setting **maxima atoms** - no default
- description: This keyword is used only together with setting **maxima atoms** (see keyword **maxima** for more details). If used without this setting, it has no effect. Setting **maxima atoms** and omitting this keyword will lead to empty output. The analysis will proceed, but a warning will be printed on the standard output. If the keyword is present, the density is searched only for maxima close to the positions of the listed atoms (see keyword **tolerance**).

4.3.3 name: axisorder

- value: string of *dim* non-repeating digits from 1 to *dim*, where *dim* is the dimension of the input density
- default: 123 ... *dim*
- description: The value of the keyword represents a permutation of the superspace axes to be used by the program. First three digits denote the real-space axes x, y, z , the rest determines the order of the axes in additional dimension. This setting is useful for composite structures, where the real-space and internal-space axes interchange their role for different subsystems. Consider for example a composite structure with the incommensurate direction along a . The most probable setting of the superspace axes will be a_1, b, c, a_2 . Thus, the fourth axis will be the additional axis of the first subsystem and simultaneously the x -axis of the second subsystem. Consequently, to analyze the first subsystem of the composite structure, use setting `axisorder 1234`, to analyze the second subsystem, use setting `axisorder 4231`.

4.3.4 name: basins

- value: `yes/no`
- default: `no`
- description: EDMA uses Bader's formalism to assign charge to atoms (Bader (1990); Section 2.3). This formalism divides the whole space into "atomic basins". The charge anywhere in the basin belongs to the atom lying in that basin. EDMA produces maps of these basins for each t , if setting `basins yes` is present in the input file. For more details about the maps of atomic basins see Section 5.4.

4.3.5 name: cell

- value: six real numbers representing $a b c \alpha \beta \gamma$
- default: compulsory keyword — no default
- description: Lattice parameters of the structure. If the output coordinates are fractional (see keyword `scale`) and the absolute values of atomic charges are not needed, the values of the lattice parameters do not influence the output.

4.3.6 name: centerofcharge

- value: `yes/no`
- default: `no`
- description: If `yes`, EDMA calculates the weighted average of coordinates of the pixels in the atomic basin of particular maximum, that have density values above some user defined limit (see keyword `chlimit`). These coordinates represent a center of charge of the atomic basin and thus an alternative position of the atom. This position can be more accurate than the position of maximal electron density if significant anharmonic movement is present. The atomic charges of individual atomic basins are calculated at the same time and written in the output.

4.3.7 name: centers - endcenters

- value: Each line contains one centering vector.
- default: no centering vectors
- description: Defines the centering vectors of the (super)space group. The components can be given both as fractions and as fractional number. For more details see keyword `symmetry - endsymmetry`.

4.3.8 name: chlimit

- value: real number between 0 and 1
- default: 0.25
- description: This keyword has effect only in combination with the setting `centerofcharge yes`. Only pixels with density $\rho > \text{chlimit} * \rho_{max}$ (ρ_{max} is the maximum density of the atomic basin) are included in calculation of the center of charge and atomic charges.

4.3.9 name: chlimlist

- value: positive real number [relative/absolute]
- default: 0.0
- description: Only maxima with integrated charge larger than `chlimlist` are listed in the output. This keyword can be used to filter out spurious maxima that are often present in the maps and that are sometimes extremely increasing the length of the output. With `chlimlist relative` the limit for the charge is calculated as a fraction of the largest charge in the map.

4.3.10 name: composition

- value: Sequence of groups elemental symbol[number of atoms in unit cell] separated by spaces. Example: Na4 V4 O10
- default: unknown composition
- description: This setting is applicable only in combination with the keyword `export` and setting `centerofcharge yes`. The manner how EDMA uses the information from `composition` depends on the value of `numberofatoms`. With `numberofatoms guess` or `numeroatoms integer` EDMA tries to assign the proper atomic type to each maximum found in the density based on the integrated charge. The largest charge is associated with the heaviest element in the list and other maxima are assigned atomic type based on the proportion of their charge to the maximum charge.

With `numberofatoms composition` EDMA assigns the atomic types to the maxima so that the composition of the resulting structure corresponds to the required composition.

If keyword `export` appears in the input list, the calculation of the center of charge is automatically turned on in order to allow the assignment of the atomic types according to the "weight" of each maximum.

Element symbol not followed by a number is equivalent to the element symbol followed by 1. The number of atoms should always correspond to the full unit cell and not to the asymmetric unit.

Hydrogen atoms in composition are always ignored, because hydrogen atoms cannot be seen as separate maxima in a normal Fourier map.

4.3.11 name: `cplist - endcplist`

- value: each line contains fractional coordinates of expected position of a critical point.
- default: compulsory keyword in combination with setting `maxima atoms` - no default
- description: This keyword is used only together with setting `criticalpoints list` (see keyword `criticalpoints` for more details). If used without this setting, it has no effect. Setting `criticalpoints list` and omitting this keyword will lead to empty output. The analysis will proceed, but a warning will be printed on the standard output. If the keyword is present, the density is searched only for critical points close to the positions of the listed coordinates (see keyword `tolerance`).

4.3.12 name: `criticalpoints`

- value: `none/all/atoms` [`simple/expanded`]
- default: `list`, if atoms are defined in `cplist - endcplist`, `none` otherwise
- description: Setting `criticalpoints all` will cause all critical points found in each t-map to be listed in the output file. Setting `criticalpoints list` results in list of the critical points near the coordinates given in the keyword `cplist - endcplist`. Use this setting if you are interested in properties of just a few important critical points, like the bond critical points. Setting `criticalpoints none` avoids any coordinates to be printed in the output. All critical points are subject to acceptance/rejection according to the keyword `plimit`. *Important:* Setting `criticalpoints all` must not be combined with setting `maxima atoms` and setting `criticalpoints list` must not be combined with setting `maxima all`. These combinations will lead to unpredictable results.

4.3.13 name: `export`

- value: a valid filename with extension `cif`, `ins` or `m40`.
- default: no file written
- description: If this keyword is present in the input, the maxima located by EDMA are written in a structure file. Depending on the extension of the file name, the structure is written in the CIF format, in the `ins` format of SHELX or in the `m40` format of the crystallographic package of Jana2000. If the expected composition is defined in the keyword `composition`, an attempt is made to assign correct atomic types to the atoms. The number of maxima that are exported can be influenced by the keywords `numberofatoms` (if you know the expected number of atoms in the unit cell), or keywords `plimit` and `chlimlist`.

Note on the INS format: EDMA does not have all information necessary for the INS format. In particular, the wavelength used for the measurement is unknown to EDMA and therefore a default $\text{MoK}\alpha$ wavelength is used.

4.3.14 name: `fullcell`

- value: `yes/no`
- default: `no`
- description: Influences the listing of the maxima. If `fullcell yes` is specified, all maxima in the unit cell are listed, otherwise only symmetry-independent maxima are given. Obviously, this is effective only if the right symmetry is provided - see keyword `symmetry - endsymmetry`.

4.3.15 name: inputfile

- value: valid specification of the file shorter or equal in length to 132 characters
- default: compulsory keyword - no default
- description: Specifies the file containing the input electron density map.

4.3.16 name: maxima

- value: none/all/atoms
- default: atoms, if atoms are defined in edmaatoms - endedmaatoms, all otherwise
- description: Setting maxima all will cause all maxima found in each t-map to be listed in the output file. Setting maxima atoms results in list of the maxima near the coordinates given in the keyword edmaatoms - endedmaatoms. This setting is suited for convenient extracting of the modulation functions for a list of atoms. Setting maxima none avoids any coordinates to be printed in the output. It can be useful if t-maps or maps of atomic basins are the only desired output. All maxima are subject to acceptance/rejection according to the keyword plimit. *Important:* Setting criticalpoints all must not be combined with setting maxima atoms and setting criticalpoints list must not be combined with setting maxima all. These combinations will lead to unpredictable results.

4.3.17 name: numberofatoms

- value: guess/composition/non-negative integer
- default: guess
- description: This keyword influences the list of atoms written in the structure file (see keyword export. Value guess will result in export of all atoms with peak density above the plimit and integrated charge above chlimlist. Value composition means that the program will take the expected number of each type of atoms from the keyword composition - see that keyword for details. An integer number tell EDMA to export so many symmetry-independent atoms that the total number of atoms in the unit cell will be just the value of numberofatoms. The exact number might slightly differ depending on the symmetry of the density and on atoms lying at special positions. As a trivial example, if the largest maximum has a multiplicity of 4, the smallest number of atoms in the exported structure will be 4, even if numberofatoms 2 were given.
numberofatoms 0 is equivalent to numberofatoms guess, and numberofatoms *negative integer* is equivalent to numberofatoms composition.

4.3.18 name: outputbase

- value: any string of 132 or less characters valid as a part of a filename
- default: compulsory keyword - no default
- description: The value of this keyword defines the base of all output files. Each output file begins with this base and is given specific ending and/or extension. For closer description of the naming conventions see Section 5.

4.3.19 name: **plimit**

- value: real number
- default: 0.0
- description: Only maxima and critical points with $\rho_{max} > \text{plimit}$ are listed in the output. This keyword can be used to filter out spurious maxima and the large amount of low-density critical points, that are often present in the MEM maps and that are sometimes extremely increasing the length of the output.

4.3.20 name: **position**

- value: **relative/absolute**
- default: **absolute**
- description: Applies only to the setting **maxima atoms** and **criticalpoints list**. If **position** is **absolute**, the coordinates in the output are related to the origin of the unit cell. With setting **position relative** the output coordinates are relative to the basic structure coordinates specified in the keyword **edmaatoms - endedmaatoms** or **cplist - endcplist**.

4.3.21 name: **projection**

- value: **yes/no**
- default: **no**; not applicable if $dim = 3$
- description: With **projection yes** EDMA projects the superspace density along all additional dimensions onto the real space and performs all the analysis on the projected density. Useful for obtaining properties of the average structure of a modulated compound.

4.3.22 name: **qvectors - endqvectors**

- value: Each line contains coordinates of one q-vector. There must be $(dim - 3)$ q-vectors between the start- and end-keyword.
- default: compulsory keyword — no default; not applicable if $dim = 3$
- description: This keyword contains the definition of the q-vectors. The order in which they are listed must correspond to the order of axes defined in the keyword **axisorder**. For example, we can have a 5D map with a standard order of axes $x_1 x_2 x_3 x_4 x_5$ and two q-vectors q_1 and q_2 corresponding to axes x_4 and x_5 . If the value of the keyword **axisorder** is **12345**, the order of the q-vectors must be q_1, q_2 . If the value of the keyword **axisorder** is **12354**, the order of the q-vectors must be q_2, q_1 .

4.3.23 name: **range**

- value: positive odd integer
- default: 7
- description: The electron density at arbitrary points is determined by spline interpolation of electron density in some neighborhood of that pixel. The keyword **range** defines the size of this neighborhood. With setting **range 7** the density of a given point is determined from the cube with side 7 pixels centered on that the point. The larger

the value, the more “global” is the interpolation, but the longer computation time is needed. In practice value 7 produces almost identical results with any higher value and value 11 is perfectly safe. The value should be odd because of some symmetry of the interpolation. Even values will work too, but they are not recommended.

4.3.24 name: scale

- value: `angstrom/fractional`
- default: `fractional`
- description: Influences the scale of the coordinates in the output. Useful to obtain the modulation functions directly in Ångstroms.

4.3.25 name: symmetry - endsymmetry

- value: each line contains definition of one symmetry operator
- default: no symmetry is assumed apart from the identity
- description: This keyword contains the complete definition of the symmetry of the input density with exception of the centering vectors. Each line contains one symmetry operator. The format of the symmetry operators corresponds to the conventions used in the International Tables for Crystallography. If the symmetry operator is $\{R|\tau\}$, then the format of each entry is:

$$\tau_1 + R_{11}x_1 + R_{12}x_2 + \dots + R_{1n}x_n \quad \tau_2 + R_{21}x_1 + R_{22}x_2 + \dots + R_{2n}x_n \quad \dots$$

Example - n-glide perpendicular to b:

$$1/2+x_1 \ -x_2 \ 1/2+x_3$$

The translation components can be given either as fractions or decimal numbers. The symmetry is not imposed on the map, and it is the user’s responsibility to provide a symmetry that is compatible with the map. The symmetry is used only for generating symmetry-independent list of maxima.

4.3.26 name: tlist - endtlist

- value: Each line contains three real numbers between 0 and 1. There must be $(dim-3)$ lines between the begin- and end-keyword.
- default: compulsory keyword - no default; not applicable if $dim = 3$
- description: Each line of this setting applies to one additional dimension in order determined by the keyword `axisorder`. Each line has format $t_{start} \ t_{end} \ t_{step}$. t_{start} defines starting t in that dimension, t_{end} defines ending t and t_{step} the steps between individual t ’s. Real-dimensional sections are calculated starting with t_{start} in steps of t_{step} as long as the generated value does not exceed t_{end} .

4.3.27 name: tmap

- value: `yes/no`
- default: `no`
- description: Turns on/off saving of the t-sections to separate files for inspection. For the description of the output files and their naming conventions see Section 5.3.

4.3.28 name: tolerance

- value: positive real number
- default: 0.15
- description: This setting is applicable only in combination with the setting `maxima atoms` and `criticalpoints list`. The critical point (including the maxima) is considered to belong to some critical point from the list defined by keyword `edmaatoms - endedmaatoms` or `cplist - endcplist` if the difference between the coordinates of the critical point and of the supposed position from the list in Å is smaller than `tolerance` in all coordinates. Tolerance should be selected larger than the largest modulation. At the same time it must not be too large to avoid confusion with possible neighboring critical points.

4.3.29 name: writem40

- value: a valid filename
- default: no file written
- description: This is an obsolete synonym for the keyword `export`. Please use `export` instead of `writem40`.

4.4 Examples of input files**4.4.1 Example 1.**

This is the simplest possible input file applicable only to the 3D densities. The output will be a list of fractional coordinates of all maxima present in the map. The filename of the output will be `example1.coo`.

```
inputfile example1.m81
outputbase example1
cell 1. 1. 1. 90. 90. 90.
```

4.4.2 Example 2.

A run of EDMA with this input file would extract all significant maxima (above $50 \text{ e}/\text{Å}^3$) from eleven *t*-sections. The structure is supposed to be a (3 + 1)D modulated structure. It could be used for a preliminary analysis of the map with unknown positions of the atoms. The output will contain only the fractional coordinates of the maxima. All eleven *t*-sections would be written to files.

```
inputfile example2.m81
outputbase example2
cell 3.128 3.128 8.245 90. 90. 120.
plimit 50.
maxima all
scale fractional
centerofcharge no
tmap yes

qvectors
0. 0. 0.2356
endqvectors
```

```
tlist
0.0 1.0 0.1
endtlist
```

4.4.3 Example 3.

This example uses most of the options EDMA offers. It leads to determination of the modulation functions of the two listed atoms of the second subsystem of the composite structure $(\text{LaS})_{1.14}\text{NbS}_2$. The modulation will be listed relative to the basic-structure coordinates in \AA and would contain 26 points:

```
inputfile LaSNbS.m81
outputbase LaSNbS
cell 5.7983 5.7972 22.9555 90. 90. 90.
tolerance 0.08
range 11
plimit 20.
chlimit 0.20
axisorder 4231
maxima atoms
position relative
scale angstroem
centerofcharge yes

qvector
1.7536 0. 0.
endqvector

tlist
0.0 1.0 0.04
endtlist

edmaatoms
La2 0.0 0.00069 0.32733 S2 0.0 0.50701 0.30005
endedmaatoms
```

4.4.4 Example 4.

This example illustrates the way EDMA can be used to obtain information about the critical points in the electron density. It leads to determination of the modulation functions of the listed atoms as well as the properties of the bond critical points between the BE and F atoms. All the coordinates will be listed in fractional coordinates:

```
inputfile ambef.m81
outputbase ambef_cp
cell 7.5281 5.8851 10.437 90. 90. 90.
tolerance 0.4
tmap no
basins no
plimit 0.3
addborder 0.2
centerofcharge yes
maxima atoms
criticalpoints list
position absolute
scale fractional
```

```

qvector
0.4794 0. 0.
endqvector

tlist
0.0 1.0 0.05
endtlist

range 9

cplist
Be-F1_cp 0.1723 0.250000 0.4033
Be-F2_cp 0.2625 0.250000 0.4758
Be-F3_cp 0.2833 0.1625 0.3993
endcplist

edmaatoms
Be 0.251406 0.250000 0.418471
F1 0.054423 0.250000 0.380154
F2 0.270298 0.250000 0.564187
F3 0.338204 0.034886 0.365746
N1 0.187086 0.250000 0.099897
N2 0.457782 0.250000 0.803255
endedmaatoms

```

4.4.5 Example 5.

The last example shows the usage of EDMA for creating a structure file in Jana2000 Format from an input electron density, typically provided by SUPERFLIP. The essential components are the symmetry operations, the `writem40` and `numberofatoms` keyword as well as composition. The input file is in `xplor` format.

```

cell 7.0192 8.4063 4.6264 90.00 108.61 90.00
centers
1/2 1/2 0
endcenters
symmetry
! spacegroup C2/m
x1 x2 x3
-x1 x2 -x3
-x1 -x2 -x3
x1 -x2 x3
endsymmetry
inputfile example_5.xplor
outputbase example_5
writem40 example_5.m40
numberofatoms 22
composition Cr P 0
maxima all
fullcell no
scale fractional
centerofcharge yes
chlimit 0.2500
chlimlist 0.1000 relative

```

Chapter 5

Description of the output

5.1 File outputbase.coo

The principal output file has the name `outputbase.coo` (see keyword `outputbase`). It consists of two parts: the header and the data. The header contains information about the input parameters and the density map. Its contents might vary depending on the input parameters, but it is self-explaining. The header of the output file of example 3 in previous section will look like this:

```
# Analysis of the density map LaSNbS_6626.m81
# Dimension of the map : 4
# Division of the map : 64 64 256 64
# Order of the axes : 4231
# Range for spline interpolation: 11
# Cell: 3.3065 5.7972 22.9555 90.0000 90.0000 90.0000
# q-vectors:
# 1.75360 0.00000 0.00000
# limits and steps of t:
# start end step
# 0.00000 1.00000 0.04000
# The center of charge and total charge of atomic basins will be computed.
# The density in the incommensurate directions is calculated from -1.300 to 1.300
# The density limit for the pixel to be accepted in the charge calculation is 0.20*Rho(max)
# The scale of the output coordinates will be angstroem
# Positions of following atoms +- 0.080 and with density over 20.000 will be searched for maxima at each t
# Coordinates are related to the input coordinates of the atoms.
# La2 0.00000 0.00069 0.32733
# S2 0.00000 0.50701 0.30005
```

The header is terminated with a blank line.

The format of the data part depends on the value of the keyword `maxima` and `criticalpoints`. If the setting `maxima atoms` or `criticalpoints list` is present in the input file, the data part is divided into blocks corresponding to individual entries in the lists of the atoms or critical points. Each line in the block corresponds to coordinates of the entry at one `t`-setting (see Example 1 below). Blocks are separated by two blank lines. The setting `maxima all` and `criticalpoints all` will divide the data part into blocks corresponding to different `t` settings. Each line in one block corresponds to one maximum or critical point found in that `t`-section (see Example 2 below). The blocks are separated by two blank lines.

Each non-blank line of the data part is either a comment (name of the atom, name of the critical point or the `t` setting) or represents one data line.

The format in which the maxima are written is:

$[t_1, t_2 \dots] x_{max} y_{max} z_{max} [x_{coc} y_{coc} z_{coc} \text{charge}] \rho_{max}$

Subscript `max` denotes coordinates of the maximum density, subscript `coc` denotes coordinates of the center of charge. The values of `t` appear only with the setting `maxima atoms`

in the input file, the coordinates of the center of charge and the total charge is present only with the setting `centerofcharge yes` in the input file.

The format of the critical points is:

$[t_1, t_2, \dots]$ x y z type rho lambda1 lambda2 lambda3 $[x_1, y_1, z_1]$ $[x_2, y_2, z_2]$ $[x_3, y_3, z_3]$ The values of t appear only with the setting `maxima atoms` in the input file. The values lambda1, lambda2 and lambda3 are the principal curvatures at the critical point. The values x_i , y_i and z_i represent components of the directions of the three principal curvatures at the critical point and are present only if the setting `criticalpoints all/list expanded` is present in the input file.

Example 1: This example corresponds to the input file of Example 4. in Section 4.4:

```
# Be
#           x           y           z           x_coc           y_coc           z_coc           charge           density           vol.
0.000      0.251366    0.249405    0.418499    0.251461    0.249391    0.418462    2.37293    26.0331    2.1166
0.050      0.251366    0.248618    0.418499    0.251453    0.248586    0.418494    2.37439    26.0322    2.1162
0.100      0.251366    0.247955    0.418499    0.251452    0.247905    0.418520    2.37260    26.0315    2.1033
.
.
.
0.950      0.251366    0.250236    0.418499    0.251466    0.250241    0.418465    2.37360    26.0334    2.1142
1.000      0.251366    0.249405    0.418499    0.251461    0.249391    0.418462    2.37293    26.0331    2.1166

# F1
#           x           y           z           x_coc           y_coc           z_coc           charge           density           vol.
0.000      0.055038    0.228896    0.376028    0.054939    0.234958    0.379807    9.49282    31.5233    14.6856
0.050      0.055064    0.229868    0.375959    0.054956    0.235398    0.379798    9.49253    31.4029    14.6916
0.100      0.055037    0.233026    0.375935    0.054938    0.237326    0.379812    9.49034    31.1313    14.6942
.
.
.
0.950      0.054975    0.230089    0.376091    0.054943    0.235962    0.379802    9.49320    31.4003    14.6912
1.000      0.055038    0.228896    0.376028    0.054939    0.234958    0.379807    9.49282    31.5233    14.6856

# F2
.
.
.

# F3
.
.
.

# N1
.
.
.

# N2
.
.
.

# Be-F1_cp
#           x           y           z           type           density           lambda1           lambda2           lambda3
0.000      0.173918    0.242703    0.402639    (3,-1)           0.5202           -3.3934           -3.3332           26.1572
0.050      0.173908    0.242333    0.402657    (3,-1)           0.5209           -3.4166           -3.3321           26.1969
0.100      0.173896    0.242753    0.402663    (3,-1)           0.5222           -3.4364           -3.3392           26.2875
.
.
```

```

.
0.950      0.173921  0.243842  0.402616  (3,-1)  0.5205  -3.3773  -3.3447  26.1830
1.000      0.173918  0.242703  0.402639  (3,-1)  0.5202  -3.3934  -3.3332  26.1572

# Be-F2_cp
#
#           x           y           z           type           density           lambda1           lambda2           lambda3
0.000      0.258758  0.259954  0.476250  (3,-1)  0.5485  -3.9674  -3.7203  26.3484
0.050      0.258763  0.258175  0.476271  (3,-1)  0.5503  -3.9866  -3.6990  26.4715
0.100      0.258745  0.255562  0.476298  (3,-1)  0.5540  -4.0280  -3.7193  26.7200
.
.
.
0.950      0.258736  0.260613  0.476253  (3,-1)  0.5490  -3.9705  -3.7211  26.3878
1.000      0.258758  0.259954  0.476250  (3,-1)  0.5485  -3.9674  -3.7203  26.3484

# Be-F3_cp
.
.
.

```

Example 2: This example illustrates the output of the run with settings `maxima all` and `centerofcharge no:`

```

# t= 0.000
0.286588  0.423447  -0.000043  1646.19
0.859758  0.423054  0.000144  1686.35
0.000002  0.922324  -0.000232  1676.74
.
.
.
0.000002  0.922324  0.999768  1676.74
0.565829  0.922539  1.000209  1670.88

# t= 0.040
0.263647  0.423486  -0.000014  1664.99
0.836763  0.423222  0.000115  1648.99
-0.020935  0.922352  -0.000231  1648.26
.
.
.
-0.020935  0.922352  0.999769  1648.26
0.545323  0.922432  1.000222  1680.50

# t= 0.080
0.238186  0.423490  0.000012  1651.24
0.812799  0.423351  0.000080  1679.12
0.523123  0.922357  0.000231  1643.32
.
.
.

```

The readers using the plotting program GNUPLOT will have already recognized that the output is designed to be easily viewed with this program. The sign # indicates comment lines, which are not interpreted by GNUPLOT, and double blank lines separate individual fields.

5.2 Files `outputbase.tmp` and `outputbase.tmc`

If setting `maxima atoms` or `criticalpoints list` is present in the input file, EDMA must store the extracted information in a temporary file in a "raw" format (`outputbase.tmp` for the maxima and `outputbase.tmc` for the critical points) and at the end of the run the

information is reordered to obtain blocks of individual atoms or critical points rather than individual t-sections. These temporary files are not automatically erased, because they can be used to obtain at least partial information for cases when the run of EDMA terminates prematurely and repeated run would take too much time. It is recommended to erase these temporary files manually from time to time.

5.3 t-maps

Sometimes it is desirable to have the opportunity to view the real-dimensional sections of the superspace density. EDMA calculates these sections and upon request (setting `tmap yes`) saves them as so called t-maps in the .m81 format. The saved files can be then viewed with program CONTOUR of the software package JANA2000. The names of the t-maps follow the pattern

`outputbase_t1-t2...tn.m81`

`t1` etc. stands for the value of t in the corresponding dimension given to two decimal places. For 4D maps only one t is present, of course. `outputbase` represents the value of the keyword `outputbase` in the input file. The t-maps contain one basic unit cell extended according to the value of the keyword `adddborder` in each non-periodic direction. Consequently, setting `adddborder` to (say) 3 will produce a t-map, which covers 7 basic unit cells in each non-periodic direction. No t-maps can be produced from ordinary 3D maps.

Note on the naming of the t-maps and maps of atomic basins (Section 5.4): The t coordinates are written in the file name with 2 decimal digits. If the real coordinates have more than two significant decimal digits, they are rounded in the filename (but not in the actual calculation). This can cause problems, if the step in t is smaller than 0.01 (see keyword `tlist - endtlist`). Two consecutive maps will then have the same name and the program will deny to write the second map with an error message

Error! Cannot open density file *mapname*

The problem can be overcome by first writing all t maps with step larger or equal to 0.01 and then, in the second run, writing another set of t-maps with a modified `outputbase`.

5.4 Atomic basins

According to Bader's formalism (Section 2.3), each point in the space can be assigned to one atom. This results in a division of the space into so called atomic basins. These basins are used to calculate the total charge of the atom and the center of charge. The atomic basins can be saved in the format .m81 (see Section 4.1). Each pixel in the map of atomic basins is assigned a number of maximum it belongs to. If such a map is viewed in the plotting program CONTOUR of the crystallographic package JANA2000, it appears in 2D sections as a mosaic of black fields (constant "density" = atomic basin of one atom) separated by lines on the border of the atomic basins, where an abrupt change of the number takes place. Due to the interpolation scheme used in CONTOUR, some borders (those with large difference between the numbers of maxima) are separated by several close-lying lines rather than by a single line. The names of the maps of basins follow the pattern

`outputbase_basins_t1-t2...tn.m81`

`t1` etc. stands for the value of t in corresponding dimension. `outputbase` represents the value of the keyword `outputbase` in the input file. In the 3D case the map of basins has the name `outputbase_basins.m81`.

For a note on the naming of the output files see Section 5.3.

Bibliography

- Bader, R. F. W. (1990), *Atoms in Molecules: A Quantum Theory*, Clarendon Press, Oxford Science Publications, Oxford.
- Coppens, P. (1997), *X-Ray Charge Densities and Chemical Bonding*, Oxford University Press, New York.
- Hirshfeld, F. (1977), 'Bonded-atom fragments for describing molecular charge densities', *Theoret. Chim. Acta* **44**, 129–138.
- Petříček, V. & Dušek, M. (2000), *The crystallographic computing system JANA2000*, Institute of Physics, Praha, Czech Republic.
- Press, W. H., Teukolsky, S. A., Vetterling, W. T. & Flannery, B. P. (1996), *Numerical Recipes in Fortran 77, Second Edition*, Cambridge University Press.
- van Smaalen, S. (1995), 'Incommensurate crystal structures', *Crystallogr. Rev.* **4**, 79–202.
- von der Linden, W., Dose, V., Fisher, R. & Preuss, R., eds (1998), *Maximum Entropy and Bayesian Methods*, Kluwer Academic Publishers, Dordrecht.