

# **2-Dimensional Finite Difference Hartree-Fock Program**

with Libxc, OpenMP and p-thread support

User's Guide

© 1996-2023 Jacek Kobus

August 2023

# Contents

<b>1</b>	<b>Description of input data</b>	<b>3</b>
1.1	Mandatory labels . . . . .	3
1.2	Optional labels . . . . .	6
<b>2</b>	<b>Program's data files</b>	<b>16</b>
<b>3</b>	<b>Running the program</b>	<b>17</b>

# 1 Description of input data

x2dhf program accepts input data that consist of separate lines which contain

- a label
- a label followed by a string of characters, integer(s) and/or real number(s)
- a string of characters, integer(s) and/or real number(s)

Real numbers can be written in a fixed-point or scientific notation. Note that

- labels and strings can be in upper or lower case,
- the compulsory labels must follow the order given below; the optional ones can be inserted anywhere between the **title** and **stop** labels,
- optional parameters are enclosed in square brackets,
- $r$  denotes a real number,  $i$  – an integer,  $c$  – a string of characters,
- an exclamation mark or a hash placed anywhere in an input line starts a comment and what follows “!” or “#” is ignored.

## 1.1 Mandatory labels

The following labels must be specified in the specified order:

### TITLE

**Format:** **title**  $c$

$c$  is any string of up to 74 characters describing the current case. This string is added as a header to a text file with a .dat extension that contains basic data identifying a given case, i.e. atomic numbers of nuclei, grid size and the number of electrons and orbital and exchange functions.

### NUCLEI

**Format:** **nuclei**  $Z_A$   $Z_B$   $R$  [ $c$ ]

Set the nuclei charges and the bond length.

$Z_A$ : nuclear charge of centre A (real)

$Z_B$ : nuclear charge of centre B (real)

$R$ : bond length (real)

$c$ : *angstrom* – the internuclear separation can be given in Ångström if this string is included (the conversion factor 0.529177249 is used)

If  $|Z_A - Z_B| < 10^{-6}$  then the molecule is considered to be a homonuclear one (see `initVariables`).

### CONFIG

**Format:** **config**  $i$

$i$ : the total charge of a system

The following cards define molecular orbitals and their occupation **in inverse order - the highest orbitals are on top and the lowest on bottom**. Note that the last orbital description card (corresponding to the deepest lying orbital) must contain the *end* label.

The possible formats are:

**Format:**  $i \ c$

$i$ : number of fully occupied orbitals of a given irreducible representation (irrep) of the  $C_{\infty v}$  group; two electrons make  $\sigma$  orbitals fully occupied and four electrons – orbitals of other symmetries

$c$ : symbol of the  $C_{\infty v}$  irrep to which the orbitals belong (*sigma*, *pi*, *delta* or *phi*)

**Format:**  $i \ c_1 \ c_2$

$i$ : number of fully occupied orbitals of a given irrep of the  $D_{\infty h}$  group

$c_1$ : symbol of the  $C_{\infty v}$  irrep to which the orbitals belong (*sigma*, *pi*, *delta* or *phi*)

$c_2$ : symbol for the inversion symmetry of the  $D_{\infty h}$  irreps (*u* or *g*)

Use this format for a homonuclear molecule unless **break** card is included.

**Format:**  $i \ c_1 \ c_2 \ [c_3 \ [c_4 \ [c_5]]]$

$i$ : number of orbitals of a given irrep of the  $C_{\infty v}$  group

$c_1$ : symbol for the  $C_{\infty v}$  irreps to which the orbitals belong (*sigma*, *pi*, *delta*, *phi*)

$c_2$ - $c_5$ : +, – or . (a dot); +/– denotes spin up/down electron and . denotes an unoccupied spin-orbital

**Format:**  $i \ c_1 \ c_2 \ c_3 \ [c_4 \ [c_5 \ [c_6]]]$

$i$ : number of orbitals of a given irrep of the  $D_{\infty h}$  group

$c_1$ : symbol for the  $C_{\infty v}$  irrep to which the orbitals belong (*sigma*, *pi*, *delta*, *phi*)

$c_2$ : symbol for the inversion symmetry of the  $D_{\infty h}$  irrep (*u* or *g*)

$c_3$ - $c_6$ : +, – or . (a dot); +/– denotes spin up/down electron and . denotes an unoccupied spin-orbital

## GRID

**Format:** **grid**  $N_\nu \ R_\infty$

An integer and a real define a single two-dimensional grid.

$N_\nu$ : the number of grid points in  $\nu$  variable

$R_\infty$ : the practical infinity

$N_\mu$  is calculated so as to make the step size in  $\mu$  variable equal to the step size in  $\nu$  variable.  $N_\nu$  and  $N_\mu$  have to meet special conditions due to the order of discretization used and the multi-threaded version of the MCSOR routine. If the conditions are not fulfilled the nearest (but smaller) appropriate values are used.

**Format:** **grid**  $N_\nu \ N_\mu \ R_\infty$

Two integers and one real define a single two-dimensional grid.

$N_\nu$ : the number of grid points in  $\nu$  variable

$N_\mu$ : the number of grid points in  $\mu$  variable

$R_\infty$ : the practical infinity

This format may be needed when interpolation between grids is attempted.

## ORBPOT

**Format:** **orbpot**  $c \ [i]$

where  $c$  a character string determining the initial source of orbitals and potentials.

The following values of  $c$  are allowed:

- *gauss* – GAUSSIAN output is used to retrieve the basis set and the molecular orbital expansion coefficients. The GAUSSIAN output is assumed to be contained in *gaussian.out* and *gaussian.pun* files.<sup>1</sup> Coulomb and exchange potentials are initialized as in the *hydrogen* case; see **prepGauss** for more details. To generate the necessary output from GAUSSIAN, you need to specify **punch=mo gfinput** in the route section of the GAUSSIAN input file. When you’ve run GAUSSIAN, you should end up with a file **fort.7** that contains the molecular orbitals. To run **x2dhf**, put the log file in the working directory as **gaussian.out** and the molecular orbitals from **fort.7** as **gaussian.pun**, which will be parsed by the program.
- *hf* – radial Hartree–Fock orbitals for the centre A and B obtained from the *qrhf* program are retrieved from disk (see `$X2DHF_DIRECTORY/hf_orbitals` directory) and Coulomb and exchange potentials are initialized unless the OED method is chosen (see routine **initHF** for details). Note that the mixing coefficients are by default set to 1; use **lcao4lda** label to change the values. Note also that the environment variable `X2DHF_DIRECTORY` must be set as the root of the *x2dhf* installation. If you don’t set `X2DHF_DIRECTORY`, the program will try to read the orbitals from the current working directory.
- *hydrogen* – molecular orbitals are formed as a linear combination of hydrogenic functions on centres *A* and *B* as defined via the **lcao** label. In the case of HF or HFS calculations Coulomb (exchange) potentials are approximated as a linear combination of Thomas–Fermi ( $1/r$ ) potentials at the two centres. If the OED method is chosen the potential function is approximated as a linear combination of  $Z_A/r_1$  and  $Z_B/r_2$  terms and the exchange potentials are set to zero.
- *lda* – radial LDA orbitals for the centre A and B obtained from the HELFEM program<sup>2</sup> are retrieved from disk (see `$X2DHF_DIRECTORY/lda_orbitals` directory) and Coulomb and exchange potentials are initialized unless the OED method is chosen (see routine **initLDA** for details). Note that the mixing coefficients are by default set to 1; use **lcao4lda** label to change the values. Note also that the environment variable `X2DHF_DIRECTORY` must be set as the root of the *x2dhf* installation. If you don’t set `X2DHF_DIRECTORY`, the program will try to read the orbitals from the current working directory.
- *noexch* – orbitals and Coulomb potentials are retrieved from disk files and exchange potentials are initialized as in *hydrogen* case; this is useful when going from DFT/HFS to HF calculations.
- *nodat* – initial orbitals and potentials are retrieved from disk files but the content of a *2dhf.input.dat* file is retrieved from a *2dhf.input.orb* (binary) file. Use this value when reading binary data generated by the earlier than 2.0 versions of the program.
- *old* – initial orbitals, Coulomb, and exchange potentials are retrieved from disk files (*2dhf.input.orb*, *2dhf.input.coul* and *2dhf.input.exch*, respectively) created in a previous run. Data defining the case are retrieved from a *2dhf.input.dat* textfile.
- *qrhf* – radial Hartree–Fock orbitals for the centre A and B obtained from the *qrhf* program<sup>3</sup> are retrieved from disk files *1dhf\_centreA.orb* and *1dhf\_centreB.orb*, respectively, and Coulomb and exchange potentials are initialized as in the *hydrogen* case (see routine **initHF** for details).

*i* is an optional parameter that is used only for cases other than *old*. To perform SCF iterations in a stable manner when both orbitals and potentials are being relaxed the potentials should correspond to the orbitals being relaxed. However, this is not the case at the very beginning of the SCF process. Therefore, at the onset of the SCF iterations

<sup>1</sup>The program has been tested on output files produced by the 94, 03, 09 and 16 versions of the GAUSSIAN system of programs.

<sup>2</sup>S. Lehtola, *Fully numerical Hartree–Fock and density functional calculations. I. Atoms*, Int. J. Quantum Chem. doi:10.1002/qua.25945

<sup>3</sup>J. Kobus and Ch. Froese Fischer, *Quasi-Relativistic Hartree–Fock program for Atoms*, to be published.

with the fairly good initial orbitals, the orbitals can be kept fixed and only Coulomb and exchange potentials should undergo relaxation. These initial relaxations of the potentials are performed until the maximum error in the orbital energy is lower than  $10^{-i}$  for *nscfextra* iterations (by default *nscfextra*=3 and *i* = 1; see `initVariables`). If *i* = 0 (the default value) then both the orbitals and potentials are relaxed at every SCF iteration.

## STOP

### Format: stop

This label indicates the end of input data.

## 1.2 Optional labels

The following additional labels can be specified in any order:

### ALTSWEEP[S]

#### Format: altsweep or altsweeps

In case of near-degenerate orbitals, i.e. when performing calculations for homonuclear molecules without enforced symmetry (no “HOMO” label), especially when the external electric field is applied, one can improve convergence by changing the direction of the SOR sweeps, i.e. using forward/backward sweeps for even/odd SCF iterations.

## BREAK

### Format: break

When this label is present homonuclear molecules are calculated in  $C_{\infty v}$  symmetry and the  $D_{\infty h}$  symmetry labels (*u* or *g*) are redundant.

## CONV

### Format: conv [*i*<sub>1</sub> [*i*<sub>2</sub> [*i*<sub>3</sub> ]]]

Sometimes the requested accuracy of a solution is set too high and cannot be satisfied on a selected grid. As a result SCF/SOR iteration process may continue in vain. To save CPU time the iterations are stopped if orbital energies or orbital norms display no improvement over the *i*<sub>2</sub> and *i*<sub>3</sub> most recent iterations, respectively (20 by default). This mechanism is activated after *i*<sub>1</sub> initial iterations (600 by default).

## DEBUG

### Format: debug [*i*<sub>1</sub> [*i*<sub>2</sub> ...] *i*<sub>40</sub>]

Up to 40 different debug flags can be set at a time. If the integer *i*<sub>*k*</sub> is present the debug flag *i*<sub>*k*</sub> is set, i.e. `idbg(ik) = 1` ( $1 \leq i_k < 999$ ). These are used to generate additional debugging information by adding the lines of the form

```
if (idbg(ik).eq.1) then
  print *, ‘‘debugging something ...’’
  ...
endif
```

## DFT

### Format: dft [*c*<sub>1</sub>] [*c*<sub>2</sub>]

$c_i$  : specifies the type of DFT exchange-correlation functional(s) to be used in Fock equations. Any of the suitable LDA, GGA, and hybrid functionals implemented in Libxc library<sup>4</sup> can in principle be used, e.g. `xc_lda_x`, `xc_lda_c-wigner`, `xc_lda_c_xalpha`, `xc_gga_c-gam`, etc; use `lxcctl list` utility to generate the list of available functionals (the utility can also be used to test the functionals, see `lxcctl help`).

For backward compatibility the following functionals can also be selected:

- `lda` – the local density approximation with the potential

$$V_X(\alpha) = -\frac{3}{2}\alpha \left(\frac{3}{\pi}\right)^{1/3} 2^{-2/3} \sum_{\sigma} \rho_{\sigma}^{1/3}$$

where  $\alpha$  is by default set to 2/3 (the Slater exchange potential). To change this value use the **xalpha** label. This functional is equivalent to `lda_x` one of the Libxc library.

If `xalpha` parameter is set to 1/3 then `lda` functional is equivalent to `xc_lda_c_xalpha` one.

- `b88` – the Becke exchange potential; this functional is equivalent to `xc_gga_x_b88` one from the Libxc library<sup>5</sup>
- `lyp` – the correlation potential of Lee, Yang and Parr; this functional is equivalent to `xc_gga_c_lyp` one from the Libxc library
- `vwn` – the correlation potential of Vosko, Wilk and Nusair; this functional is equivalent to `lda_c_vwn` one from the Libxc library<sup>6</sup>

When the bare label is present and the method selected is HF then the exchange contributions (LDA, B88) and the correlation contributions (LYP and VWN) to the total energy are calculated upon completion of the SCF iterations.

## DENSTHLD

**Format:** `densthld` [  $r$  ]

$r$  : sets the value of `dens_threshold` parameter used in the Libxc library to avoid returning NaN values (the default value is defined in `initVariables` subroutine).

## FASTSCF

**Format:** `fastscf`

This label causes the SCF process to automatically skip relaxing orbitals (and also Coulomb and exchange potentials) for which the orbital energy threshold has been reached. This is a very desirable behaviour when (and the default one) when the OED method is used to calculate a set of virtual orbitals.

## FEFIELD

**Format:** `feffield`  $r$

$r$ : a strength of an external static electric field directed along the internuclear axis (in atomic units)

## FERMI

---

<sup>4</sup>Susi Lehtola, Conrad Steigemann, Micael J.T. Oliveira, and Miguel A.L. Marques, *Recent developments in Libxc - A comprehensive library of functionals for density functional theory*, Software X 7, 1 (2018)

<sup>5</sup>Cf. the results of test sets `he/set-09`, `he/set-10` and `he/set-11`.

<sup>6</sup>In case the Hartree-Fock orbitals of the FH molecule are used to calculate the correlation contribution the functional `vwn` gives -0.699579 while `xc_lda_c_vwn` gives a slightly smaller value, namely, -0.704211.

**Format:** `fermi` [ $r_A$  [ $r_B$ ]]

When this label is present the Fermi nuclear charge distribution is used. Optional parameters  $r_A$  and  $r_B$  define the atomic masses (in atomic mass units, amu) of nuclei A and B. If omitted the corresponding values are taken from the table of atomic masses compiled by Wapstra and Audi (see `blk.data`).

## FIXORB

**Format:** `fixorb` [ $i_1$  [ $i_2 \dots$ ]  $i_{40}$ ]

This label is used to specify orbitals to be kept frozen during SCF/SOR process (they are not being renormalized nor orthogonalized during the process).  $i_1, i_2, \dots$  are the numbers of these orbitals as they appear on the program's listing, i.e. their order is reversed to that used when defining the electronic configuration (see the `config` card). Up to 40 different orbitals can be set at a time. Use the bare label to keep all orbitals frozen.

## FIXPOT

**Format:** `fixpot`

If this label is present then all Coulomb and exchange potentials are kept frozen during the SCF/SOR process.

## GAUSS

**Format:** `gauss` [ $r_A$  [ $r_B$ ]]

When this label is present the Gauss nuclear charge distribution is used. Optional parameters  $r_A$  and  $r_B$  define the atomic masses (in amu) of nuclei A and B. If omitted, the corresponding values are taken from the table of atomic masses compiled by Wapstra and Audi (see `blk.data`).

## HOMO

**Format:** `homo`

This label is used to impose explicitly  $D_{\infty h}$  symmetry upon orbitals of homonuclear molecules to improve SCF/SOR convergence.

## INOUT

**Format:** `inout`  $c_1$   $c_2$

The x2dhf program can be compiled to support calculation using three different combinations of integer/real data types: i32 (4-byte integers, 8-byte reals), i64 (8-byte integers, 8-byte reals) and r128 (8-byte integers, 16-byte reals). Strings  $c_1$  and  $c_2$  determine the combination appropriate for the format of input and output data, respectively, and each string can be i32, i64 or r128.

To facilitate the exchange of binary data generated on machines of different architectures or using different compilers additional formats are available, namely i32f, i64f or r128f. They can be used to export/import data in the formatted instead of the (default) unformatted form.

## INTERP

**Format:** `interp` [ $\text{iord}_{\nu}^{\text{orb}}$   $\text{iord}_{\mu}^{\text{orb}}$   $\text{iord}_{\nu}^{\text{coul}}$   $\text{iord}_{\mu}^{\text{coul}}$   $\text{iord}_{\nu}^{\text{exch}}$   $\text{iord}_{\mu}^{\text{exch}}$ ]

Use this label to change the grid between separate runs of the program. The interpolation order in the  $\nu$  and  $\mu$  directions used for the orbitals, Coulomb potential and exchange potentials can be set with the six optional parameters. Supported orders for the interpolation are 2, 4 (default), 6, and 8.

## KINPOT



**Format: kinpot**

Upon the completion of the SCF process the Pauli,  $v^P(\mathbf{r})$ , and von Weizsäcker kinetic potentials,  $v^W(\mathbf{r})$ , are calculated along the internuclear axis and written to out4kinpot.dat file together with the total density,  $\rho(\mathbf{r})$ ,  $\nabla^2\rho(\mathbf{r})$ ,  $\tau(\mathbf{r})$  (the kinetic energy density) and  $\tau^W(\mathbf{r}) = |\nabla\rho(\mathbf{r})|^2/(8\rho(\mathbf{r}))$ ,

**LCAO****Format: lcao [ i ]**

If the source of orbitals is declared as *hydrogen* then this card must be present. In such a case the initialization of each of the orbitals has to be defined in terms of the linear combination of atom centred hydrogen-like functions. For each orbital include a card of the following format (make sure that the order of orbitals should match the order specified under the **config** label):

**Format:**  $c_A \ n_A \ l_A \ \zeta_A \ c_B \ n_B \ l_B \ \zeta_B \ [ \ i_1 \ [ \ i_2 \ ] ]$

where

$c_A$  – relative mixing coefficient for a hydrogenic orbital on the  $Z_A$  centre (real),

$n_A$  – its principal quantum number (integer)

$l_A$  – its orbital quantum number (integer)

$\zeta_A$  – the effective nuclear charge if  $i = 1$  (default) or a screening parameter if  $i = 2$  (real)

$c_B$  – relative mixing coefficient for a hydrogenic orbital on the  $Z_B$  centre (real),

$n_B$  – its principal quantum number (integer)

$l_B$  – its orbital quantum number (integer)

$\zeta_B$  – the effective nuclear charge if  $i = 1$  (default) or a screening parameter if  $i = 2$  (real)

$i_1$  – (integer) if set to 1 (the default) indicates that this orbital should be initialized as a linear combination of hydrogenic functions and not taken from a disk orbital file (integer). When the source of orbitals is declared as *old* and the orbital and Coulomb potential data files contain fewer functions than defined by *config* card this flag can be used to indicate which orbitals are missing in orbital data files and require initialization. This can be useful when generating virtual orbitals for a potential formed from the already given set of converged orbitals. Indicate such orbitals by setting this parameter to 0.

$i_2$  – (integer) if set to 0 (1, the default) indicates that this orbital will not contribute (will contribute) to the electron density when evaluating DFT functionals. It allows therefore to generate virtual orbitals within the DFT calculations since a local potential can be built from a given subset of orbitals that are kept frozen during the SCF process (see **fixorb** label).

The mixing coefficients are normalized so that  $|c_A| + |c_B| = 1$

**LCAO4LDA**

**Format: lcao4lda [  $c_A$  [  $c_B$  ] ]**

If the source of orbitals is declared as *lda* then use this card to adjust the mixing coefficients (by default  $c_A = c_B = 1.0$ ); see label **LCAO**.

**LM**

**Format: lm  $i_1 \ i_2 \ [ \ i_3 \ i_4 \ ] \ \dots \ [ \ i_{N-1} \ i_N \ ]$**

For closed-shell systems the off-diagonal Lagrange multipliers are set to zero. However, for some such systems the SCF/SOR convergence can be improved if for some pair(s) of orbitals the non-zero values of the Lagrange multipliers are used (see, for example, the o2, li2-elprop or f2-elprop test suits).

## MCSOR|MCSOR-O

**Format:** `mcsor—mcsor-o` [ $i_1$  [ $i_2$  [ $i_3$  ] ]

Selects the MCSOR method for solving the Poisson equations for orbitals and the SOR method – for Coulomb and exchange potentials.  $i_1$  parameter can be used to set the number of threads employed. The default value depends on the parallelization method used; see `initVariables`. Use  $i_2$  and  $i_3$  to set the value of the MCSOR micro relaxation sweeps during a single SCF cycle for orbitals and potentials, respectively (by default  $2 = i_3 = 10$ ).

When the MCSOR method is selected via this label then the single-threaded version of the multi-colour SOR method is chosen by default to relax orbitals. When OPENMP/PTHREAD/TPOOL directive is present the parallelized version of MCSOR is employed.<sup>7</sup> Coulomb and exchange potentials are relaxed in parallel using the single-threaded SOR method. However, when the MCSOR-CE label together with OPENMP directive is used also the potentials are relaxed using the parallelized version of the MCSOR routine (in case PTHREAD/TPOOL directive is used the SOR method must be used to avoid nested parallel regions).

Labels MCSOR-O and MCSOR-CE were added to force the usage of the MCSOR selectively for orbitals and Coulomb/exchange potentials, respectively. See `relaxDriver` for more details.

b

## MCSOR-CE

**Format:** `mcsor-ce` [ $i$  ]

When OPENMP directive is used this label selects the MCSOR method for solving the Poisson equations for Coulomb and exchange potentials and, if  $i > 0$ , sets the number of threads used. In case of PTHREAD/TPOOL directive the SOR method is used (see `coulExchSORPT`).

## METHOD

**Format:** `method`  $c$

Select the type of calculation.

- $c$ : HF – the Hartree-Fock method
- $c$ : DFT – the Hartree-Fock method with the  $X\alpha$  exchange potential ( $\alpha = 2/3$ ); see the `dft` label to choose another exchange or correlation potential
- $c$ : HFS – the Hartree-Fock-Slater method (Hartree-Fock with the  $X\alpha$  exchange potential) with the optimum value of the  $\alpha$  parameter (see `src/commons.F90`)
- $c$ : OED – One Electron Diatomic ground and excited states can be calculated for the Coulomb potential in the prolate spheroidal coordinates (default). It is also possible to specify the Coulomb and Kramers-Henneberger potentials in cylindrical coordinates (see the `poth3`, `potkh`, `potharm2`, `potharm3` labels, respectively). When more than one orbital is specified calculations are carried out as if in the case of a multielectron system.<sup>8</sup>
- $c$ : TED – Two Electron Diatomic ground and excited states can be calculated for the Hooke's atom (harmonium). See `POTHOKE` label.
- $c$ : SCMC – the Hartree-Fock method with  $X\alpha$  exchange where the  $\alpha$  parameter is calculated according to the self-consistent multiplicative constant method<sup>9</sup>

<sup>7</sup>The TPOOL directive should be used together with the MUTEX one (see `x2dhfctl -t -X`).

<sup>8</sup>In this type of calculations convergence rates differ greatly between orbitals. Therefore, if for a given orbital the orbital energy threshold is reached it is being frozen.

<sup>9</sup>V. V. Karasiev and E. V. Ludeña, *Self-consistent multiplicative constant method for the exchange energy in density functional theory*, Phys. Rev. A **65** (2002) 062510. doi:10.1103/PhysRevA.65.062510

## MMOMENTS

**Format:** `mmoments`

Calculate the multipole moments for the converged orbitals relative to the geometric centre and the centre of mass. The total charge densities at (0,0,-R/2) and (0,0,+R/2) are also evaluated.

## MULTIPOL

**Format:** `multipol r [ i ]`

- r*: if  $r > 0$  multipole moment expansion coefficients are recalculated when the maximum error in orbital energy is changed by  $r$  (the default value is 1.15; see `initVariables`). To suppress recalculation of the coefficients set  $r$  to a negative real number. This is useful when generating potentials from a set of fixed orbitals, e.g. from GAUSSIAN orbitals.
- i*: the number of terms in the multipole expansion used to calculate the boundary values for potentials ( $2 \leq i \leq 8$  and the default is 4).

## OMEGA

**Format:** `omega  $\omega_{orb}$  [  $\omega_{pot}$  ]`

One or two real numbers setting over-relaxation parameters for relaxation of orbitals and potentials; see also **OMEGAOPT**. The negative value of  $\omega_{orb}/\omega_{pot}$  indicates that its value should be set to a near optimum value obtained from a semiempirical formula (see `initCBlocks` and `setOmega` for details).

## OMEGAOPT

**Format:** `omegaopt [ i [  $r_{orb}$  [  $r_{pot}$  ] ] ]`

The optional integer parameter can be set to 1 (default) or 2. In the former case the over-relaxation parameters are based on tests performed for the FH molecule on 157x229/45 grid and scaled down a bit (by 0.979 and 996 for orbitals and potentials, respectively). In the latter, the values are calculated from a formula for the optimal value of the over-relaxation parameter in case the model Poisson equation is discretized using the second-order stencil.

**This label should only be used for test purposes.**

where  $c$  a character string determining the initial source of orbitals

## ORDER

**Format:** `order c`

where  $c$  is a character string defining the ordering of mesh points: “col-wise” – natural column-wise, “middle” – ‘middle’ type of sweep (default), “row-wise” – natural row-wise, “row-wise” – reversed natural column-wise (see `mesh` routine for details)

## PLOT

**Format:** `plot [  $i_1$  [  $i_2$  ] ]`

Orbital values are written in formatted form to be plotted. For  $i_1$  equal to 1, 2 or 3 values  $(z_j, x_k, f(z_j, x_k))$ ,  $(\nu_j, \mu_k, f(\nu_j, \mu_k))$  or  $(\eta_j, \xi_k, f(\eta_j, \xi_k))$  are written, respectively, and every  $i_2^{\text{th}}$  value in each variable is selected, i.e.  $j = 1, N_\nu, i_2$  and  $k = 1, N_\mu, i_2$ . By default  $i_1 = 1$  and  $i_2 = 2$ . The values in the tail region are omitted if they are msmaller than a given threshold (by default  $10^{-10}$ ; see `plotThreshold` variable in `initVariables` routine).

## POTGSZ

**Format:** `potgsz`

When the OED method is chosen then this label selects a model potential due to Green, Sellin and Zachor.<sup>10</sup> For a given atom this potential produces HF-like orbitals and it has been found useful in finding decent starting orbitals for diatomic molecules as well.<sup>11</sup> Note, however, that the superposition of atomic potentials (SAP) guess<sup>12</sup> (keyword: `potsap`) is much more accurate, as it is not restricted to a simple analytic form.

## POTGSZG

**Format:** `potgszg`

When the OED method is chosen then this label selects a model potential due to Green, Sellin and Zachor and the Gauss nuclear charge distribution. For a given atom this potential produces HF-like orbitals but it has been found useful in finding decent starting orbitals for any molecular system.

## POTHOKE

---

<sup>10</sup>A. E. S. Green, D. L. Sellin and A. S. Zachor, *Analytic Independent-Particle Model for Atoms*, Phys. Rev. **184** (1969) 1-9. doi:10.1103/PhysRev.184.1

<sup>11</sup>J. E. Whalen and A. E. S. Green, *Analytic Independent Particle Model for Molecules*, Am. J. Phys. **40**, (1972) 1484. doi:10.1119/1.1986874; K. J. Miller and A. E. S. Green, *Energy levels and potential energy curves for H<sub>2</sub>, N<sub>2</sub>, and O<sub>2</sub> with an independent particle model*, J. Chem. Phys. **60**, (1974) 2617. doi:10.1063/1.1681415; T. Sawada, P. S. Ganas, and A. E. S. Green, *Elastic scattering of electrons from N<sub>2</sub>*, Phys. Rev. A **9**, (1974) 1130. doi:10.1103/PhysRevA.9.1130

<sup>12</sup>S. Lehtola, *Assessment of Initial Guesses for Self-Consistent Field Calculations. Superposition of Atomic Potentials: Simple yet Efficient*, J. Chem. Theory Comput. **15**, (2019) 1593-1604. doi:10.1021/acs.jctc.8b01089

**Format: pothooke  $k$** 

When the HF method is chosen then this label selects the potential of the Hooke atom:  $-\frac{1}{2}k^2(r_1^2 + r_2^2)$  ( $k = 1/2$  by default), where  $r_1$  and  $r_2$  denote the distances of the two electrons from the center A and thus the nuclear charges should be set accordingly, e.g.  $Z_1 = 2$  and  $Z_2 = 0$ .

**POTHARM2****Format: potharm2  $k$** 

When the OED method is chosen then this label selects a two-dimensional model potential of the form  $-k^2(x^2 + y^2)$ , where  $k$  is a strength of the potential (real).

**POTHARM3|EXTRACUL[E]****Format: potharm3|extracul[e] [ $k$ ]**

When the OED method is chosen then this label selects a three-dimensional model potential of the form  $-k^2(x^2 + y^2 + z^2)$ , where  $k$  is a strength of the potential (real,  $k = 1/2$  by default).

**POTCOUL2****Format: potcoul2  $m$   $a$   $V_0$** 

When the OED method is chosen then this label selects a two-dimensional model potential of the form  $-V_0/\sqrt{a^2 + x^2 + y^2}$ . The following parameters can be set

$m$  : magnetic quantum number of a state (integer)

$a$  : width of the model potential (real)

$V_0$  : depth of the model potential (real)

**POTCOUL3****Format: potcoul3  $m$   $a$   $V_0$** 

When the OED method is chosen then this label selects a two-dimensional model potential of the form  $-V_0/\sqrt{a^2 + x^2 + y^2 + z^2}$ . The following parameters can be set

$m$  : magnetic quantum number of a state (integer)

$a$  : width of the model potential (real)

$V_0$  : depth of the model potential (real)

In order to get the hydrogen Coulomb potential set  $a = 0.0$  and  $V_0 = 1.0$ . Set  $a > 0$  and  $V > 0$  to choose its smoothed variant.

**POTKH****Format: potkh  $m$   $\varepsilon$   $\omega$  [ $a$  [ $V_0$  [ $N$ ]]]**

When the OED method is chosen then this label selects the Kramers-Henneberger potential. The following parameters can be set

$m$  : magnetic quantum number of a state being calculated (integer)

$\varepsilon$  : laser intensity (real)

$\omega$  : laser cycle frequency (real)

$a$  : original (before averaging over one laser cycle) soft-core potential width (a positive real number, by default  $a = 1.0$ )

$V_0$  : original soft-core potential depth (by default  $V_0 = 1.0$ )

$N$  : number of intervals in the Simpson quadrature (an integer,  $N = 1000$  by default)

**POTSAP**

**Format: potsap**

When the OED method is chosen, this label selects a pretabulated, screened atomic exchange potential that has been found to reproduce results in molecular calculations.<sup>13</sup> The potentials of the present implementation<sup>14</sup> are from highly accurate finite element calculations carried out with the HELFEM program.

**PRINT****Format: print** [  $i_1$  [  $i_2 \dots$  ]  $i_{40}$  ]

Up to 40 different printing flags can be set at a time. If the integer  $i_k$  is encountered the printing flag  $i_k$  is set, i.e.  $\text{iprint}(i_k) = 1$  ( $1 \leq i_k < 999$ ). These are used to generate additional printouts by adding the lines of the form

```
if (iprint(ik).eq.1) then
  print *, 'printing something ...'
  ...
endif
```

Set

- $i_1 = 110$  to print a total radial density relative to the centre A along the internuclear axis ( $-R_\infty \leq z \leq -R/2$ )
- $i_2 = 111$  to print a total radial density relative to the centre B along the internuclear axis ( $R/2 \leq z \leq R_\infty$ )

See `inputData` for a list of available flags.

**PRTEVERY****Format: prtevery**  $i_1$   $i_2$ 

Routine `pmtx` can be used to output two-dimensional arrays in a tabular row-wise form with every  $i_1$ -th row and  $i_2$ -th column being printed (by default every 10th row and column is selected)

**SCF****Format: scf** [  $i_1$  [  $i_2$  [  $i_3$  [  $i_4$  [  $i_5$  ] ] ] ] ]

- $i_1$ : maximum number of scf iterations (default 1000); to skip the scf step set  $i_1$  to a negative integer,
- $i_2$ : every  $i_2$  SCF iterations orbitals and potentials are saved to disk (default 20). If  $i_2 = 0$  functions are saved on disk upon completion of the SCF process. If  $i_2 < 0$  functions are never written to disk,
- $i_3$ : if the maximum error in orbital energy is less than  $10^{-i_3}$  for *nscfextra* iterations (3 by default; see `initVariables`), then the SCF process is terminated ( $i_3 = 10$  by default),
- $i_4$ : if the maximum error in orbital norm is less than  $10^{-i_4}$  for *nscfextra* iterations (3 by default; see `initVariables`), then SCF process is terminated ( $i_4 = 10$  by default),
- $i_5$ : the verbosity of the output during SCF process
  - $i_5 = 1$  – the orbital energy, the difference between its current and previous value, the normalization error, and the (absolute) value of the largest overlap integral between the current orbital and all the lower-lying ones of the same symmetry (the value is zero for the lowest orbitals of each symmetry) is printed for every orbital in every SCF iteration

<sup>13</sup>S. Lehtola, *Assessment of Initial Guesses for Self-Consistent Field Calculations. Superposition of Atomic Potentials: Simple yet Efficient*, J. Chem. Theory Comput. **15**, (2019) 1593-1604. doi:10.1021/acs.jctc.8b01089

<sup>14</sup>S. Lehtola, to be published.

- $i_5 = 2$  – the orbital energy, the difference between its current and previous value and the normalization error is printed for the worst converged orbital in energy (first line) and norm (second line) in every SCF iteration (default)
- $i_5 = 3$  – the orbital energy, the difference between its current and previous values and the normalization error is printed for the worst converged orbital in energy (first line) and norm (second line) every  $i_2$  iterations. Printing of “... multipole moment expansion coefficients (re)calculated ...” communicate is suppressed

The total energy is printed every  $i_2$  iterations.

## SLOWEXCH

### Format: **slowexch**

This label causes the SCF process to relax the exchange potential for a given pair of orbitals twice during an SCF cycle.

## SOR

### Format: **sor** [ $i_1$ [ $i_2$ ]]

Selects the SOR method for solving the Poisson equations for orbitals and potentials (default) and changes the value of the micro SOR relaxation sweeps during a single SCF cycle for orbitals ( $i_1$ ) and potentials ( $i_2$ ); by default  $i_1 = i_2 = 10$ .

## SOR4ORB

### Format: **sor4orb** [ $i_1$ [ $i_2$ ]]

Selects the SOR method for solving the Poisson equations for orbitals and changes the value of the SOR micro ( $i_1$ ) and macro ( $i_2$ ) relaxation sweeps during a single SCF cycle for orbitals; by default  $i_1 = 10, i_2 = 1$ .

## SOR4POT

### Format: **sor4pot** [ $i_1$ [ $i_2$ ]]

Selects the SOR method for solving the Poisson equations for potentials and changes the value of the SOR micro ( $i_1$ ) and macro ( $i_2$ ) relaxation sweeps during a single SCF cycle for potentials; by default  $i_1 = 10, i_2 = 1$ .

## TAIL

### Format: **tail**

This label causes the examination of the tail regions of the orbitals, i.e. the regions where the  $\mu$  variable has its maximum value, to find how fast the orbitals decay. These values can help choose a safe value of  $R_\infty$ .

## XALPHA

### Format: **xalpha** $\alpha$

This label allows to change the  $\alpha$  parameter of the LDA potential (see the HFS/DFT method); by default  $\alpha$  is set to 2/3. To set this parameter to zero use a very small but nonzero value, e.g. **1.0e-15**.

## 2 Program's data files

There are several standard names used by the program to keep track of its input and output disk files. Normally the program writes out the data in the course of computations and upon its completion into the following disk files:

`2dhf_output.dat` (a text (ASCII) file) containing the title of a case, the time and date of its commencement, the number of mesh points, the internuclear distance, the charges of nuclei and the number of orbitals, electrons, and exchange potentials (see `writeDisk` for details),

`2dhf_output.orb` (a binary file) containing molecular orbitals (in the order specified by the input data following `config` label) followed by their normalization factors, orbital energies, Lagrange multipliers, and multipole moment expansion coefficients (see `write`),

`2dhf_output.coul` (a binary file) containing corresponding Coulomb potentials and

`2dhf_output.exch` (a binary file) containing all exchange potentials if the `exchio` [`in-one`|`in-many`] `out-one` card is present

These files can be used to restart a given case or run another with slightly modified parameters. If `orbpot old` card is present orbitals are retrieved from `2dhf_input.orb` file, Coulomb potentials from `2dhf_input.coul` and exchange potentials from `2dhf_input.exch` file.



### 3 Running the program

To simplify the usage of the program, the xhf script (see tests/xhf) is provided to facilitate handling of the disk files. The command xhf can be invoked with one, two or three parameters. There are two basic modes of its usage:

```
xhf file1 file2
runs x2dhf reading input data from file1.data file and writing text data describing the case
into file2.dat file and binary data with orbitals and potentials into file2.orb, file2.coul
and file2.exch files.
```

```
xhf file1 file2 file3
runs x2dhf reading input data from file1.data and initial orbitals and potentials from
file2.dat, file2.orb, file2.coul and file2.exch files and writing resulting data into
file3.dat, file3.orb, file3.coul and file3.exch files.
```

If, for example, we have the following two data files:

- be.data

```
TITLE Be R_inf=35.0 bohr R = 2.3860 bohr
METHOD hf
NUCLEI 4.0 0.0 2.386
CONFIG 0
      2 sigma end
grid 169 35.0
orbpot hydrogen
lcao
  1.0 2 0 4.0 0.0 1 0 9.0
  1.0 1 0 4.0 0.0 1 0 9.0
SCF 300 10 4 8 1
stop
```

- be-1.data

```
TITLE Be R_inf=35.0 bohr R = 2.3860 bohr
METHOD hf
NUCLEI 4.0 0.0 2.386
CONFIG 0
      2 sigma end
grid 169 35.0
orbpot old
lcao
  1.0 2 0 4.0 0.0 1 0 9.0 0
  1.0 1 0 4.0 0.0 1 0 9.0 0
SCF 3000 10 8 10 1
stop
```

then

```
xhf be be-1
```

starts and performs the first 300 SCF iterations. And

```
xhf be-1 be-1 be-2
```

must be used to continue the calculations. In order to converge the SCF process even better increase the convergence parameters (see the **scf** label) and use the following command

```
xhf be-1 be-2 be-1
```

Use `testctl list` to see dozens of examples of the x2dhf usage; try `testctl help|-h|--help` to get more details.

xhf can be used to perform a number of additional tasks; see `xhf help|-h|--help`.