

Jacek Kobus

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

User's Guide

version 2.1

Contents

1	Description of input data	3
1.1	Mandatory labels	3
1.2	Optional labels	7
1.3	Deprecated labels	16
2	Examples of input data	19
3	Program's data files	32
4	How to run the program?	32
5	Useful hints	33

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 extension `dat` 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 angstrom units 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 `setDefault`s).

CONFIG

Format: `config i`

i: the total charge of a system

The following cards define molecular orbitals and their occupation. **Note that the last orbital description card 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 σ 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 c1 c2`

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

c₁: symbol of the $C_{\infty v}$ irrep to which the orbitals belong (*sigma*, *pi*, *delta* or *phi*)

c₂: 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 c1 c2 [c3 [c4 [c5]]]`

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

c₁: symbol for the $C_{\infty v}$ irreps to which the orbitals belong (*sigma*, *pi*, *delta*, *phi*)

c₂-c₅: +, – or . (a dot); +/– denotes spin up/down electron and . denotes an unoccupied spin orbital

Format: `i c1 c2 c3 [c4 [c5 [c6]]]`

- 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_ν : the number of grid points in ν variable

R_∞ : the practical infinity

N_μ is calculated so as to make the step size in μ variable equal to the stepsize in ν variable. N_ν and N_μ have to meet special conditions and if they 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_ν : the number of grid points in ν variable

N_μ : the number of grid points in μ variable

R_∞ : the practical infinity

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

ORBPOT

Format: orbpot c

where c a character string determining the initial source of orbitals and potentials. Its allowed values are:

- *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.

- *gauss* – GAUSSIAN output is used to retrieve exponents and expansion coefficients of (uncontracted) molecular orbitals (it is assumed that the output is contained in gaussian.out and gaussian.pun files) and Coulomb and exchange potentials are initialized as in the *hydrogen* case; see **prepGauss** for more details.
- *gauss-c* – GAUSSIAN customized output is used to retrieve exponents and expansion coefficients of molecular orbitals (it is assumed that the output is contained in gaussianc.out file) and Coulomb and exchange potentials are initialized as in the *hydrogen* case; see **prepGaussCust** for details.
- *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 program¹ 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).
- *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.

STOP

Format: stop

This label indicates the end of input data.

¹J.Kobus and Ch. Froese Fischer, *Quasi-Relativistic Hartree-Fock program for Atoms*, to be published.

1.2 Optional labels

The following additional labels can be specified in any order:

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_1 [i_2 [i_3]]]`

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_2 and i_3 most recent iterations, respectively (20 by default). This mechanism is activated after i_1 initial iterations (600 by default).

DEBUG

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

Up to 40 different debug flags can be set at a time. If the integer i_k is present the debug flag i_k is set, i.e. $\text{idbg}(i_k) = 1$ ($1 \leq i_k < 999$). 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_1] [c_2]`

c_1 : specifies the type of DFT exchange potential to be used in Fock equations
– $c_1 = lda$ – 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 α is by default set to 2/3 (the Slater exchange potential). To change this value use the **xalpha** label.

– $c_1 = b88$ – the Becke exchange potential

c_2 : selects the type of correlation potential to be used in Fock equations

– $c_2 = lyp$ – the correlation potential of Lee, Yang and Parr

– $c_2 = vwn$ – the correlation potential of Vosko, Wilk and Nusair

When the bare label is present and the method selected is HF then the exchange contributions (LDA, B88, PW86 and PW91) and the correlation contributions (LYP and VWN) to the total energy are calculated upon completion of the SCF iterations.

EXCHIO

Format: **exchio** c_1 c_2

These parameters specify how exchange potentials are to be read/written and manipulated (stored in random access memory). The program always keeps all orbitals and Coulomb potentials in the memory. If computer resources are adequate all exchange potentials can also be kept there. However, during the relaxation of a particular orbital only a fraction of exchange potentials is needed. Thus all exchange potentials can be kept on disk as separate files (named fort.31, fort.32, ... during a run) and only relevant ones are being retrieved when necessary.

The possible values of c_1 and c_2 are *in-one*, *in-many*, *out-one* and *out-many*. Possible combinations are

- **exchio in-many out-many** – read exchange potentials as separate files and write them back as separate files
- **exchio in-one out-many** – read all exchange potentials in one file but write them out as separate files
- **exchio in-many out-one** – read all exchange potentials separately but write them out as a single file
- **exchio in-one out-one** – read and write exchange potentials in the form of a single file (same as $i_2 = 3$); this is the default

FEFIELD

Format: **feffield** r

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

FERMI

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 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. 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.

FIXCOUL

Format: `fixcoul`

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

FIXEXCH

Format: `fixexch`

If this label is present then all 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 in order 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); see src/Makefile.am for details. 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.

In order to facilitate exchange of binary data generated on machines of different architectures or using different compilers additional formats are available, namely i32f, i64f or r128f which allow to export/import data in formatted instead of unformatted form.

INTERP

Format: interp

Use this label to change the grid between separate runs of the program. Note that only a number of grid points in one of the variables or R_∞ can be changed at a time.

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 ζ_A c_B n_B l_B ζ_B i_1 [i_2]

where

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

n_A – its principle quantum number (integer)

l_A – its orbital quantum number (integer)

ζ_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 principle quantum number (integer)

l_B – its orbital quantum number (integer)

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

i_1 – set to 1 to indicate 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 *hydrogen* these flags are ignored. 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 already given set of converged orbitals.

In case when the source of orbitals is declared as *old* and all orbitals are already defined this flag can be used in DFT calculations to generate virtual orbitals for some sort of a local potential build from a given subset of orbitals that are kept frozen during the SCF process (see **fixorb** label)

i_2 – a number of successive over-relaxations for a given orbital (integer); if omitted is set to 10

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

MCSOR

Format: **mcsor** [i_1 [i_2]]

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

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 an optimum value of the α parameter (see **blk--data.inc** for details)

- 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 Krammers-Henneberger potentials in cylindrical coordinates (see the **poth3** and **potkh** labels, respectively). When more than one orbital is specified calculations are carried out as if in the case of a multielectron system.²
- c*: SCMC – the Hartree-Fock method with $X\alpha$ exchange where the α parameter is calculated according to the self-consistent multiplicative constant method³

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 **setDefault**s). When the maximum error is less than r the coefficients are recalculated at least every i_2 SCF iterations as defined by the SCF label. 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*: number of terms in the multipole expansion used to calculate boundary value for potentials ($2 \leq i \leq 8$ and the default is 4).

OMEGA

Format: **omega** ω_{orb} [ω_{pot}]

One or two real numbers setting overrelaxation parameters for relaxation of orbitals and potentials. 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}]]]

Optional integer parameter can be set to 1 (default) or 2. In the former case rather conservative (safe) ω values are chosen (this is equivalent to using the

²In 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.

³V.V.Karasiev and E.V.Ludenia, *Self-consistent multiplicative constant method for the exchange energy in density functional theory*, Phys. Rev. A **65** (2002) 062510.

omega card with the negative values of the parameters). In the latter case somewhat better values are chosen but faster convergence is to be expected only when good initial estimates of orbitals and potentials are available or when calculations with fixed orbitals or potentials are performed. The near-optimal ω values obtained from a semi-empirical formula are scaled down to produce final values used by the program. The default values of the scaling factors for the orbital and potential overrelaxation parameters (0.986 and 0.997, respectively) can be changed by setting r_{orb} and r_{pot} to their desired values.

ORDER

Format: `order [i]`

An integer defining the ordering of mesh points: 1 – natural column-wise, 2 – ‘middle’ type of sweep (default), 3 – natural row-wise, 4 – reversed natural column-wise (see `mesh` routine for details)

POTGSZ

Format: `potgsz`

When the OED method is chosen then this label selects a model potential due to Green, Sellin and Zachor.⁴ For a given atom this potential produces HF-like orbitals but it was found useful in finding decent starting orbitals for any molecular system.

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 was found useful in finding decent starting orbitals for any molecular system.

POTH3

Format: `poth3 m a V0`

When the OED method is chosen then this label selects a two-dimensional model potential of the form $-V_0/\sqrt{a^2 + r^2}$ (see `kh.c` for details). The following parameters can be set

m : magnetic quantum number of a state (integer)

⁴A.E.S.Green, D.L.Sellin and A.S.Zachor, *Analytic Independent-Particle Model for Atoms*, Phys. Rev. **184** (1969) 1-9.

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 ε ω [a [V_0 [N]]]`

When the OED method is chosen then this label selects the Krammers-Henneberger potential (see routine `kh.c` for details). The following parameters can be set

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

ε : laser intensity (real)

ω : 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)

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. `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} than the SCF process is terminated (10 by default),
- i_4 : if the maximum error in orbital norm is less than 10^{-i_4} than SCF process is terminated (10 by default),
- i_5 : the level of 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
 - $i_5 = 2$ – 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

Total energy is printed every i_2 iterations.

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 SOR relaxation sweeps during a single SCF cycle for orbitals (i_1) and potentials (i_2); by default $i_1 = i_2 = 10$.

XALPHA

Format: `xalpha α`

This label allows to change the α parameter of the LDA potential (see the HFS/DFT method); by default α is set to 2/3.

1.3 Deprecated labels

The following labels have been replaced by others but are supported for backward compatibility with the previous version of the input data:

INITIAL (deprecatd, use **ORBPOT** and **LCAO** instead)

Format: `initial i_1 [i_2 [i_3]]`

i_1 : determine the initial source of orbitals and potentials:

- $i_1 = 1$ – molecular orbitals are formed as a linear combination of hydrogenic functions on centres A and B (see the description of i_3 for further details); 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 method OED 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
- $i_1 = 2$ – GAUSSIAN output is used to retrieve exponents and expansion coefficients of (uncontracted) molecular orbitals (it is assumed that the output is contained in gaussian.out and gaussian.pun files) and Coulomb and exchange potentials are initialized as in $i_1 = 1$ case; see **prepGauss** for more details
- $i_1 = 3$ – GAUSSIAN customized output is used to retrieve exponents and expansion coefficients of molecular orbitals (it is assumed that the output is contained in gaussianc.out file) and Coulomb and exchange potentials are initialized as in $i_1 = 1$ case; see **prepGaussCust** for details
- $i_1 = 5$ – 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.

- $i_1 = 6$ – orbitals and Coulomb potentials are retrieved from disk files and exchange potentials are initialized as in $i_1 = 1$ case (convenient when going from HFS to HF calculations)
- $i_1 = 11$ – radial Hartree-Fock orbitals for the centre A and B are retrieved from disk files 1dhf_inputA.orb and 1dhf_inputB.orb, respectively, and Coulomb and exchange potentials are initialized as in the *hydrogen* case (see routine `initHF` for details).
- $i_1 = 55$ – 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 older versions of the program.

i_2 : specifies how exchange potentials are to be read/written and manipulated (stored in random access memory). The program always keeps all orbitals and Coulomb potentials in the memory. If computer resources are adequate all exchange potentials can also be kept there. However, during the relaxation of a particular orbital only a fraction of exchange potentials is needed. Thus all exchange potentials can be kept on disk as separate files (named fort.31, fort.32, ... during a run) and only relevant ones are being retrieved when necessary.

- $i_2 = 0$ – read exchange potentials as separate files and write them back as separate files
- $i_2 = 1$ – read all exchange potentials in a file but write them out as separate files
- $i_2 = 2$ – read all exchange potentials separately but write them out as a single file
- $i_2 = 3$ – read and write exchange potentials in the form of a single file (default)

i_3 : if $i_1 = 1$ then this parameter must be set to 1 or 2 (if omitted it is set to 1). In such a case the initialization of each of the orbitals has to be defined in terms of the linear combination of atom centered 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 principle quantum number (integer)
 l_A – its orbital quantum number (integer)
 ζ_A – the effective nuclear charge if $i_3 = 1$ or a screening parameter if $i_3 = 2$ (real)
 c_B – relative mixing coefficient for a hydrogenic orbital on the Z_B centre (real),
 n_B – its principle quantum number (integer)
 l_B – its orbital quantum number (integer)
 ζ_B – the effective nuclear charge if $i_3 = 1$ or a screening parameter if $i_3 = 2$ (real)
 i_1 – set to 1 to freeze the orbital during SCF; otherwise set to 0 (integer)
 i_2 – a number of successive over-relaxations for a given orbital (integer); if omitted is set to 10
 For other values of i_1 than 1 the orbital cards can be omitted but then the i_3 parameter must be set to 0.

FIX

Format: `fix [i_1 [i_2 [i_3]]]`

If i_1 , i_2 or i_3 are set to 1 then orbitals, Coulomb potentials or exchange potentials, respectively, are kept frozen during the SCF/SOR process (the respective default values are 0, 0 and 2). If $i_3 = 2$ then exchange potentials are relaxed only once during an SCF cycle. i_2 and i_3 cannot be set to 1 if hydrogenic orbitals are used to initiate the orbitals.

OMEGA

Format: `omega`

ω_{orb}

ω_{pot}

Two real numbers setting over-relaxation parameters for relaxation of orbitals and potentials. The negative value of a given parameter indicates that its value should be set to a near optimum value obtained from a semiempirical formula (see `initCBlocks` and `setomega` for details).

2 Examples of input data

1. ^2S ground state of the Th^{+89} one-electron system (see examples/oed/th+89/th+89-1s.lst).

```
TITLE Th+89 point/finite nucleus R = 2.5
METHOD OED
NUCLEI 90.0 0.0 2.0
CONFIG 89
  1 sigma + end
GRID 169 2.5
orbpot hydrogen
lcao
  1.0 1 0 90.0 0.0 1 0 1.0 0
SCF 30 10 8
omega 1.80 1.80
! fermi 232.0 0.0
STOP
```

2. First excited ^2S state of the Th^{+89} one-electron system (see examples/oed/th+89/th+89_2s.lst).

```
TITLE Th+89 point/finite nucleus R = 2.5
METHOD OED
NUCLEI 90.0 0.0 2.0
CONFIG 88
  1 sigma +
  1 sigma + end
GRID 169 2.5
orbpot old
lcao
  1.0 2 0 90.0 0.0 1 0 1.0 0
  1.0 1 0 90.0 0.0 1 0 1.0 1 ! 1s orbital must be kept frozen
SCF 30 10 8
omega 1.80 1.80
!fermi 232.0 0.0
STOP
```

3. Hartree-Fock ground state of the beryllium atom calculated in two consecutive steps (see examples/be/be.lst and examples/be/be-1.lst).

```
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  0
  1.0   1 0 4.0      0.0   1 0 9.0  0
SCF 300 10 4 4 1
! note that omega for orbitals and potentials is set automatically
! omega -1.92 -1.87
stop
```

```
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 8 1
stop
```

4. Hartree-Fock ground state energy of the hydrogen molecule (see examples/h2/h2.lst).

```
title H2: R = 2.0au !!!!!
method hf
!nuclei 1.0 1.0 1.4
nuclei 1.0 1.0 2.0
!homo
!break
config 0
  1 sigma g end
grid 169 40.0
orbpot hydrogen
lcao
  1.0 1 0 1.0 1.0 1 0 1.0 0
scf 400 10 16 16
omega 1.95 -1.87
stop
```

5. Hartree-Fock ground state of the BF molecule (see examples/bf/bf_init2.lst).

6. HF calculations for the lowest 3P state of the carbon atom (see examples/c/c.lst).

```

title 3P C    R = 2.386
method hf
nuclei  6.0  0.0  2.386
config  0
  1 pi    + . +
  1 sigma
  1 sigma  end
grid 169 30.0
orbpot hydrogen
lcao
  1.0  2 1 5.0      0.0  1 0 9.0  0
  1.0  2 0 6.0      0.0  1 0 9.0  0
  1.0  1 0 6.0      0.0  1 0 9.0  0
scf 400 20 12 12
omega 1.80 1.87
stop

```

7. HF calculations for the lowest 2P state of the C^+ ion (see examples/c+/c+.lst).

```

title 1P C+    R = 2.386
method hf
nuclei  6.0  0.0  2.386
config  1
  1 pi    +
  1 sigma
  1 sigma  end
grid  169 30.0
orbpot hydrogen ! orpot old ! using 3P results
lcao
  1.0  2 1 5.0      0.0  1 0 9.0  0
  1.0  2 0 6.0      0.0  1 0 9.0  0
  1.0  1 0 6.0      0.0  1 0 9.0  0
scf 500 20 7
!omega 1.82 1.87
stop

```

8. HF calculations for the lowest state of the C_2 molecule (see examples/c2/c2a.lst and examples/c2/c2b.lst).

```
TITLE C2      R = 2.358
METHOD HF
NUCLEI  6.0 6.0 2.358
homo
!break
CONFIG  0
  1 pi    u
  1 sigma u
  1 sigma g
  1 sigma u
  1 sigma g end
GRID  169 40.0
orbpot gauss
! just a couple of dozen iterations to start with
SCF 50 10 10 10 1
! note the small value of overrelaxation parameter for orbitals
! often initial scf/sor iterations require slower convergence rates
omega 1.25 1.85
STOP
```

```
TITLE C2      R = 2.358
METHOD HF
NUCLEI  6.0 6.0 2.358
homo
!break
CONFIG  0
  1 pi    u
  1 sigma u
  1 sigma g
  1 sigma u
  1 sigma g end
GRID  169 40.0
orbpot old
SCF 500 10 10 10 1
! try a negative value for the second overrelaxation parameter to get
  its more or less optimal value
```

```
omega 1.75 -1.85
STOP
```

9. HF calculations for the lowest state of the N_2 molecule (see examples/n2/n2.lst).

```
TITLE N2      R = 2.068
METHOD HF
NUCLEI  7.0 7.0 2.068
homo
CONFIG  0
  1 pi    u
  1 sigma g
  1 sigma u
  2 sigma g
  1 sigma u end
GRID 169 40.0
orbpot hydrogen
lcao
  0.5   2 1 5.0    0.5   2 1 5.0   0
  1.5   2 1 5.0    0.0   2 1 5.0   0
  0.5   2 0 7.0   -0.5   2 0 7.0   0
  0.5   2 0 7.0    0.5   2 0 7.0   0
  0.5   1 0 7.0    0.5   1 0 7.0   0
  0.5   1 0 7.0   -0.5   1 0 7.0   0
SCF 2000 10 10 10 2
omega 1.65 1.85
STOP
```


10. HF calculations for the lowest state of the F_2 molecule (see examples/f2/f2.lst).

```
title F2      R = 2.668  (1.4118449A)
method hf
nuclei  9.0 9.0 2.668
homo
config  0
1 pi    g
1 pi    u
1 sigma g
1 sigma u
1 sigma g
1 sigma u
1 sigma g end
grid 169 50.0
!initial  2 3 1
orbpot gauss
lcao
0.5   2 1 5.0    0.5   2 1 5.0   0
0.5   2 1 5.0   -0.5   2 1 5.0   0
0.5   2 1 6.0    0.5   2 1 6.0   0
0.5   2 0 8.0   -0.5   2 0 8.0   0
0.5   2 0 8.0    0.5   2 0 8.0   0
0.5   1 0 9.0   -0.5   1 0 9.0   0
0.5   1 0 9.0    0.5   1 0 9.0   0
scf 2000 10 10 10 1
conv 1000
omega  1.94 -1.95
stop
```

11. A series of HF calculations for the the FH in external static electric field.

(a) no external field (see examples/fh/fh-0.lst)

```
title FH      R = 1.7328
method hf
nuclei  9.0 1.0 1.7328
config  0
    1 pi
    3 sigma end
grid 223 150.0
orbpot hydrogen
lcao
1.0  2 1 7.0    0.0  2 1 5.0  0
1.0  2 1 7.0    0.0  2 0 8.0  0
1.0  2 0 8.0    0.0  1 0 9.0  0
1.0  1 0 9.0    0.0  1 0 9.0  0
scf 2000 10 13 13 1
!fefield +0.0001
omega 1.90 -1.85
stop
```

(b) field strength -0.0001 a.u. (see examples/fh/fh-m1.lst)

```
title FH      R = 1.7328
method hf
nuclei  9.0 1.0 1.7328
config  0
    1 pi
    3 sigma end
grid 223 150.0
orbpot old
lcao
1.0  2 1 7.0    0.0  2 1 5.0  0
1.0  2 1 7.0    0.0  2 0 8.0  0
1.0  2 0 8.0    0.0  1 0 9.0  0
1.0  1 0 9.0    0.0  1 0 9.0  0
scf 2000 10 13 13 1
fefield -0.0001
omega 1.90 -1.85
stop
```

(c) field strength -0.0002 a.u. (see examples/fh/fh-m2.lst)

```
title FH      R = 1.7328
method hf
nuclei  9.0 1.0 1.7328
config  0
    1 pi
    3 sigma end
grid 223 150.0
orbpot old
lcao
1.0  2 1 7.0    0.0  2 1 5.0  0
1.0  2 1 7.0    0.0  2 0 8.0  0
1.0  2 0 8.0    0.0  1 0 9.0  0
1.0  1 0 9.0    0.0  1 0 9.0  0
scf 2000 10 13 13 1
fefield -0.0002
omega 1.90 -1.85
stop
```

(d) field strength +0.0001 a.u. (see examples/fh/fh-p1.lst)

```

title FH      R = 1.7328
method hf
nuclei  9.0 1.0 1.7328
config  0
    1 pi
    3 sigma end
grid 223 150.0
!interp
orbpot old
lcao
1.0  2 1 7.0    0.0  2 1 5.0  0
1.0  2 1 7.0    0.0  2 0 8.0  0
1.0  2 0 8.0    0.0  1 0 9.0  0
1.0  1 0 9.0    0.0  1 0 9.0  0
scf 2000 10 13 13 1
fefield +0.0001
omega 1.90 -1.85
stop

```

(e) field strength 0.0002 a.u. (see examples/fh/fh-p2.lst)

```

title FH      R = 1.7328
method hf
nuclei  9.0 1.0 1.7328
config  0
    1 pi
    3 sigma end
grid 223 150.0
orbpot old
lcao
1.0  2 1 7.0    0.0  2 1 5.0  0
1.0  2 1 7.0    0.0  2 0 8.0  0
1.0  2 0 8.0    0.0  1 0 9.0  0
1.0  1 0 9.0    0.0  1 0 9.0  0
scf 2000 10 13 13 1
fefield +0.0002
omega 1.90 -1.85
stop

```

12. DFT calculations with LDA and LYP functionals (see examples/dft/be-1.lst and examples/dft/be-2.lst).

```
title Be DFT R_inf=25.0 bohr  R = 2.3860 bohr
method  dft
dft b88 ! lyp
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  0
  1.0   1 0 4.0      0.0   1 0 9.0  0
scf 1000 10 8 8 1
! note that omega for potentials is set automatically
omega 1.80 -1.82
stop
```

```
TITLE Be DFT R_inf=25.0 bohr  R = 2.3860 bohr
METHOD  dft
dft b88 lyp
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 1000 10 8 8 1
! note that omega for potentials is set automatically
omega 1.80 -1.82
stop
```

13. Two lowest states of the 2D hydrogenic harmonic potential (see examples/oed/h3/-h3-1.lst and examples/oed/h3/h3-2.lst).

```
title Hydrogen in 2D harmonic potential: ground sigma state
method OED
nuclei 1.0 1.0 2.0
poth3 0 0.0 1.0
homo
config 0
  1 sigma g + end
grid 269 50.0
!inout i32 i32f
orbpot hydrogen
lcao
  1.0 1 0 1.0 1.0 1 0 1.0 0
scf 1000 10 10 10 1
omega 1.92 -1.87
stop
```

```
title Hydrogen in 2D harmonic potential: 1st excited sigma state
method OED
nuclei 1.0 1.0 2.0
poth3 0 0.0 1.0
homo
config 0
  1 sigma u +
  1 sigma g + end
grid 269 50.0
!inout i32 i32f
orbpot hydrogen
lcao
  1.0 2 1 1.0 0.0 1 0 1.0 0
  1.0 1 0 1.0 1.0 1 0 1.0 1
scf 1000 10 10 10 1
omega 1.92 -1.87
stop
```

14. Lowest state of the Krammers-Henneberger potential (see examples/oed/kh/kh.lst).

```
title Hydrogen in 2D Krammers-Henneberger potential: ground sigma state
method OED
nuclei 1.0 1.0 2.0
potkh 0 5.0 1.0 1.0 1.0 500 ! m eps w a V0 N
homo
config 0
  1 sigma g + end
grid 269 50.0
initial 1 3 1
  1.0 1 0 1.0 1.0 1 0 1.0 0
scf 1000 10 10 10 1
omega 1.92 -1.87
stop
```

15. SCMC ground state calculations for the beryllium atom (see examples/scmc/be-scmc.lst).

```
TITLE Be SCMC R_inf=35.0 bohr R = 2.3860 bohr
METHOD scmc
NUCLEI 4.0 0.0 2.386
CONFIG 0
  2 sigma end
GRID 91 35.0
orbpot hydrogen
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 1000 10 8 8 1
omega 1.82 -1.82
stop
```

3 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

`fort.31`, `fort.32`, ... (binary files) each containing the exchange potential required for a particular pair of orbitals if the **exchio in-many [out-one|out-many]** or **exchio [in-one|in-many] out-many** 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 (or `fort.31`, `fort.32`, ... files, if **exchio in-many [out-one|out-many]**). Note that there is only one set of `fort` files.

4 How to run the program?

In order 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, `be.data` file contains input data for the beryllium atom (see Example 3) then

```
./xhf be be-1
```

starts and performs the first 300 SCF iterations. Type

```
./xhf be-1 be-1 be-2
```

to continue 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
```

In addition, the `xhf` script can be used to perform the following tasks:

```
./xhf stop
```

creates `stop_x2dhf` file in a current directory to stop a running program

```
./xhf mkgauss filename
```

creates symbolic links `gaussian.out` and `gaussian.pun` to files `filename.out` and `filename.pun`, respectively (see e.g. Example 5).

```
./xhf rmgauss
```

removes `gaussian.out` and `gaussian.pun` files from a current directory

```
./xhf clean
```

removes `*.[dat|orb|coul|exch]` files

Use `./xhf -h|help` to get the complete list of available options.

5 Useful hints

1. The program should be easy to use provided you can start a calculation for a specific system. You should not encounter any serious problems when the system contains atoms from the first two rows of the Periodic Table. Then even the rough hydrogenic

estimates of the orbitals should prove adequate and after the initial couple of dozen of iterations a smooth convergence should set in.

If, however, a system contains more than 15-20 electrons the initial estimates of the orbitals have to be good enough to avoid divergences. Then, you have to choose the parameters of the hydrogenic orbitals carefully or perform the finite basis set calculations using the Gaussian94 to provide the initialization data for orbitals (see Example 5).

One can also use HFS method to produce initial estimates of orbitals and Coulomb potentials. For example, to start calculations for the neon atom one can use the following input data:

This input produces good enough HFS initial estimates so that the calculations can be continued at the HF level with the corresponding input:

One can also use HF method with some model potential, e.g. the model potential of Green, Sellin, Zachor (label POTGSZ).

2. At the very beginning set the maximum number of SCF iterations to something between 20 and 50 and/or impose crude convergence criteria for the orbital energy and normalization.
3. In case of convergence problems try to perform calculations on a sparser grid. For example, the $[61 \times 79/30]$ grid is sufficient to check the quality of the initial data for the Ne_2 system.
4. Choose smaller values of the relaxation parameters ($1.7 \leq \omega \leq 1.85$) to avoid divergences in the first few dozens of SCF iterations (rarely the values as small as 1.2 may be needed). Subsequently the parameters should be increased to their (near) optimum value (see the **omega** label and Example 8).

It is possible to set ω_{pot} to its near-optimal value by calculating it from a semiempirical formula; see the **omega** label. As a rule of thumb the optimal value of the orbital relaxation parameter is somewhat smaller and, by default, is obtained by scaling the ω_{pot} value by 0.98 (see **setDefault**s).

5. How to stop the program gracefully during a lengthy calculation without killing the process and interrupting disk read/write operations? All you have to do is to create a (zero length) file named **stop_x2dhf** in a working directory by typing **./xhf stop** (you can also use the Unix **touch** command to this end). The program stops whenever this file is detected upon the completion of a current orbital/potential relaxation.