

---

# **DoNOF Documentation**

***Release 0.0***

**Mario Piris and Ion Mitxelena**

**Jan 14, 2021**



## OVERVIEW:

<b>1</b>	<b>Download and Installation</b>	<b>3</b>
<b>2</b>	<b>License and Citing DoNOF</b>	<b>5</b>
<b>3</b>	<b>Contributing and Contact</b>	<b>7</b>
<b>4</b>	<b>Getting Started</b>	<b>9</b>
4.1	Running script . . . . .	10
<b>5</b>	<b>Electron-pairing-based NOF for Multiplets</b>	<b>13</b>
5.1	PNOF5 . . . . .	14
5.2	PNOF6 . . . . .	14
5.3	PNOF7 . . . . .	15
5.4	PNOF7s . . . . .	15
<b>6</b>	<b>NOF-MP2</b>	<b>17</b>
<b>7</b>	<b>Input Options</b>	<b>19</b>
7.1	Basis Set . . . . .	19
7.2	&INPRUN . . . . .	19
7.3	&NOFINP . . . . .	21
7.3.1	Number of total iterations . . . . .	21
7.3.2	Type of calculation . . . . .	21
7.3.3	HARTREE-FOCK . . . . .	21
7.3.4	PNOF Selection . . . . .	22
7.3.5	Convergence criteria in NOF calculation . . . . .	22
7.3.6	Options for the orbital optimization program (ID method) . . . . .	23
7.3.7	Options for perturbative calculations . . . . .	24
7.3.8	Restart options for GAMMA, C, diagonal F, and nuclear coordinates . . . . .	24
7.3.9	Output options . . . . .	25
7.3.10	Options related to orthonormality of Natural Orbitals . . . . .	26
7.3.11	Options related to frozen coordinates in geometry optimization . . . . .	26
7.3.12	Options for optimization program . . . . .	27
7.4	Additional Notes . . . . .	27
7.4.1	Geometry Optimization . . . . .	27
7.4.2	Dependencies . . . . .	28
7.4.3	Dissociation . . . . .	28
7.4.4	Symmetry . . . . .	29
7.4.5	WFN file . . . . .	29
7.4.6	MLD file . . . . .	29
7.4.7	Numerical Precision . . . . .	29

<b>8</b>	<b>Examples</b>	<b>31</b>
8.1	Single-point calculation . . . . .	31
8.2	Perturbative correction . . . . .	32
8.3	Geometry Optimization . . . . .	33
8.4	Convergence . . . . .	33

DoNOF is an open-source implementation of natural-orbital-functional-based methods for quantum chemistry.



The DoNOF computer program is designed to solve the energy minimization problem of a natural-orbital-functional (NOF) that describes the ground-state of an  $N$ -electron system at absolute zero temperature. The program includes the NOFs developed in the Donostia quantum chemistry group, namely PNOFi [ $i=3-7$ ]. The solution is established by optimizing the energy functional with respect to the occupation numbers and to the natural orbitals, separately.



## DOWNLOAD AND INSTALLATION

DoNOF is open-source, so you can download the single source file from GitHub: <https://github.com/DoNOF>

In the following, we describe in detail the procedure to install DoNOF in your computer. First download the source file (donof.f90) from GitHub, or simply clone this repository in your computer. Then, enter the new directory and type “make” to create the executables for DoNOF. Intel and GNU Fortran compilers can be used to create a serial and mpi executables.

The program depends on the lapack library so you must also download the ‘lapack.f’ file from our repository. If you already have this library installed on your computer and prefer to use it, you don’t need the ‘lapack.f’ file, but you do need to incorporate the necessary options into the Makefile for a successful DoNOF build.

An example of a simple Makefile is given below:

```
#####  
→ #####  
#                               Makefile for DoNOF program (Date: January 2021)                               ↵  
→ #  
#####  
→ #####
```

```
SFLAGS = -i8 -r8 -fpp -O2
```

```
# You can use -O3 with recent versions of the Intel Fortran compiler
```

```
F90 = ifort $(SFLAGS)
```

```
MPIF90 = mpiifort -DMPI $(SFLAGS)
```

```
#
```

```
SFLAGSG = -fdefault-integer-8 -fdefault-real-8 -fdefault-double-8 -cpp -ffpe-summary=none -O3
```

```
F90g = gfortran $(SFLAGSG)
```

```
MPIF90g = mpif90 -DMPI $(SFLAGSG)
```

```
#
```

---

```
all: serial mpi serialg mpig
```

---

```
serial: $(F90) -o DoNOF.x donof.f90 lapack.f
```

```
mpi: $(MPIF90) -o DoNOFmpi.x donof.f90 lapack.f
```

```
serialg: $(F90g) -o DoNOFg.x donof.f90 lapack.f
```

```
mpig: $(MPIF90g) -o DoNOFmpig.x donof.f90 lapack.f
```

---





## LICENSE AND CITING DONOF

Copyright by Mario Piris (2009), IPR registered under Basque Government and Spanish Ministry ECD, Registration number 01/2020/360.

GPL (version 3) is applied to DoNOF. For more info check <https://www.gnu.org/licenses/licenses/gpl-3.0.html>

The use of this software is restricted solely to academic institutions and for academic purposes. This software may not be sold or any other commercial use made of it without the express written permission of the author.

By citing DoNOF you are encouraging development and exposing the software package.

Publication about DoNOF:

M. Piris, I. Mitxelena, “DoNOF: an open-source implementation of natural-orbital-functional-based methods for quantum chemistry”, *Comp. Phys. Comm.* 259, 107651 (2021)

It should be cited as:

```
@article{Piris2021,  
  author = {Piris, Mario and Mitxelena, Ion},  
  doi = {10.1016/j.cpc.2020.107651},  
  journal = {Computer Physics Communications},  
  pages = {107651},  
  title = {DoNOF: an open-source implementation of natural-orbital-functional-based methods for quantum chemistry},  
  volume = {259},  
  year = {2021}  
}
```

You can find a preprint corresponding to this publication in:

<https://arxiv.org/abs/2004.06142>



## CONTRIBUTING AND CONTACT

The DoNOF code is open-source, and thus we appreciate any contribution to the code, such as

- Ideas of missing features
- Ideas for improving documentation
- Bugs in the code or output
- Errors in the documentation
- New tutorials

It is preferable to get in touch through GitHub, at <https://github.com/DoNOF/DoNOFsw>. As usual, you have to FORK DoNOF, make the corresponding modifications in your own FORK, and finally do a PULL REQUEST to the main/master repository of DoNOF. Remember that you can clone the master DoNOF repository, but since you have read-only permission, you cannot PUSH to the master branch.

Otherwise, you can contact us by e-mail to [DoNOFsw@gmail.com](mailto:DoNOFsw@gmail.com)



## GETTING STARTED

All PNOF options have default values (see “Input Options” section), so for a given system defined in the \$INPRUN part of the input, it only remains necessary to put:

```
&NOFINP /
```

An input for single-point energy calculation of Hydrogen atom with minimal basis set reads as:

```
&INPRUN RUNTYP='ENERGY' MULT=1 ICHARG=0 /
$DATA
H atom: STO-2G basis set calculation

H  1.0  0.00  0.00  0.00
S   2
   1      1.309756377      0.4301284983
   2      0.2331359749      0.6789135305

$END
&NOFINP /
```

or even simpler if the file containing the basis set (basis-name.bas) is in the directory where the calculation is performed (\$PWD), in \$HOME/DoNOFsw/basis/, \$HOME/DoNOF/basis/ directories, or if you indicate the site where it is located, that is, PATH/basis-name.bas. In that case an input for single-point energy calculation of Hydrogen atom with sto-2g.bas basis-set file reads as:

```
&INPRUN RUNTYP='ENERGY' MULT=1 ICHARG=0 /
$DATA
H atom: STO-2G basis set calculation
sto-2g
H  1.0  0.00  0.00  0.00
$END
&NOFINP /
```

In each calculation many files are generated. Imagine the previous input corresponds to hydrogen.inp, then we will obtain after a single-point calculation:

hydrogen.out → file containing general output data corresponding to the NOF calculation

hydrogen.gcf → file containing info needed to restart any calculation from the output of this one

hydrogen.wfn → file containing wave-function info for AIMPAC program, among others.

## 4.1 Running script

You can find in DoNOF GitHub repository the scripts we usually employ to run the program, including those necessary after Intel or GNU compilation.

A very simple serial running script may read as:

```
#!/bin/csh -f

if (-f $1.gcf) mv -f $1.gcf GCF

if (-f $1.fra) mv -f $1.fra FRAG

nice +18 $PATH-TO-EXECUTABLE/DoNOF.x < $1.inp > $1.out

if (-f FCHK) mv -f FCHK $1.fchk

if (-f MLD)then
  cp -f MLD $1.mld
  if (-f XYZ)then
    cat MLD XYZ > $1.mld
    rm -f XYZ
  endif
  rm -f MLD
endif

if (-f WFN) mv -f WFN $1.wfn

if (-f APSG) mv -f APSG $1.pun

if (-f 1DM) mv -f 1DM $1.1dm

if (-f fort.14) mv -f fort.14 $1.1dm

if (-f 2DM) mv -f 2DM $1.2dm

if (-f fort.15) mv -f fort.15 $1.2dm

if (-f N2DM) mv -f N2DM $1.n2dm

if (-f CJK) mv -f CJK $1.cjk

if (-f CND) mv -f CND $1.cnd

if (-f Tijab) mv -f Tijab $1.2mp

if (-f FRAG) mv -f FRAG $1.fra

if(-f GCFe)then
  mv -f GCFe $1.gcf
  rm -f GCF
else if(-f GCF)then
  mv -f GCF $1.gcf
endif

if (-f CGGRAD) mv -f CGGRAD $1.cgo
```

(continues on next page)

(continued from previous page)

```
if (-f CGM) rm -f CGM  
if (-f BFST) rm -f BFST  
if (-f fort.1) rm -f fort.1
```





## ELECTRON-PAIRING-BASED NOF FOR MULTIPLETS

The non-relativistic electronic energy can be written as an explicit functional of the 1- and 2-order reduced density matrices (RDM)

$$E_{el}[\Gamma, D] = \sum_{ik} \Gamma_{ki} h_{ki} + \sum_{ijkl} D_{kl,ij} \langle kl | ij \rangle$$

where, if there is no any dependence on spin coordinates, then RDMs correspond to a mixed quantum state (ensemble) that allows all possible spin projections for a given total spin quantum number  $S$ .

$$\Gamma_{ki} = \frac{1}{2S+1} \sum_{M_s=-S}^S \langle SM_s | \hat{a}_k^\dagger \hat{a}_i | SM_s \rangle$$

$$D_{kl,ij} = \frac{1}{2(2S+1)} \sum_{M_s=-S}^S \langle SM_s | \hat{a}_k^\dagger \hat{a}_l^\dagger \hat{a}_j \hat{a}_i | SM_s \rangle$$

In practical applications, we approximate the 2RDM in terms of the occupation numbers, so this leads to an approximate NOF, namely

$$E_{el}[\{n_i, \phi_i\}] = \sum_i n_i \mathcal{H}_{ii} + \sum_{ijkl} D[n_i, n_j, n_k, n_l] \langle kl | ij \rangle$$

In our electron-pair-based NOF the orbital space is divided into disjoint orbital subspaces. This allows us to conserve correct spin symmetry, both total spin and spin projection. Also, since the spin-restricted formalism can be employed even for  $S > 0$ , the formula to compute energy gradients holds. Then it is straightforward to compute equilibrium geometries of non-singlet systems. See details at PRA 100, 032508 (2019) and “Analytic gradients for spin multiplets in natural orbital functional theory” (available also in arXiv:2005.02333 [physics.chem-ph]). In general, the NOF for spin-multiplets can be written as

$$E = \sum_{g=1}^{\frac{N_{II}}{2}} E_g + \sum_{g=\frac{N_{II}}{2}+1}^{N_{\Omega}} \mathcal{H}_{gg} + \sum_{f,g=1; f \neq g}^{N_{\Omega}} E_{fg}$$

$$E_g = 2 \sum_{p \in \Omega_g} n_p \mathcal{H}_{pp} + \sum_{q, p \in \Omega_g} \Pi_{qp} \mathcal{K}_{pq}, \quad \Omega_g \in \Omega_{II}$$

$$E_{fg} = \sum_{p \in \Omega_f} \sum_{q \in \Omega_g} [n_q n_p (2\mathcal{J}_{pq} - \mathcal{K}_{pq}) - \Phi_q \Phi_p \mathcal{K}_{pq}]$$

DoNOF contains several approximations that lead to the NOFs named as PNOFi [i=3-7] (in the input file we choose one or another by setting IPNOF=i). For more info see IJQC 113, 620 (2013), and also the references given below.

## 5.1 PNOF5

(JCP 134, 164102, 2011)

PNOF5 is an independent-pair model. It is a pure N-representable NOF, since it corresponds to an APSG wavefunction with fixed phases (see JCP 139, 234109, 2013).

$$\begin{aligned}
 E_{el}^{pno5} &= \sum_{g=1}^{N/2} E_g + \sum_{f \neq g}^{N/2} E_{fg} \\
 E_{fg} &= \sum_{p \in \Omega_f} \sum_{q \in \Omega_g} [n_q n_p (2\mathcal{J}_{pq} - \mathcal{K}_{pq})] \\
 E_g &= \sum_{p \in \Omega_g} n_p (2\mathcal{H}_{pp} + \mathcal{J}_{pp}) + \sum_{q, p \in \Omega_g, q \neq p} \Pi_{qp}^g \mathcal{L}_{pq} \\
 \Pi_{qp}^g &= \begin{cases} -\sqrt{n_q n_p}, & p = g \text{ or } q = g \\ +\sqrt{n_q n_p}, & p, q > N/2 \end{cases}, \quad \delta_{q\Omega_g} = \begin{cases} 1, & q \in \Omega_g \\ 0, & q \notin \Omega_g \end{cases}
 \end{aligned}$$

## 5.2 PNOF6

(JCP 141, 044107, 2014)

In the case of PNOF6, the only difference with respect to PNOF5 relies on the interaction between electrons that belong to different electron pairs. For PNOF6 this interaction is defined as

$$E_{fg} = \sum_{p \in \Omega_f} \sum_{q \in \Omega_g} E_{pq}^{int} = \sum_{p \in \Omega_f} \sum_{q \in \Omega_g} [(n_q n_p - \Delta_{qp}) (2\mathcal{J}_{pq} - \mathcal{K}_{pq}) + \Pi_{qp} \mathcal{L}_{pq}]$$

where

$\Delta_{qp}$	$\Pi_{qp}$	<i>Orbitals</i>
$e^{-2S} h_q h_p$	$-e^{-S} (h_q h_p)^{\frac{1}{2}}$	$q \leq F, p \leq F$
$\frac{\gamma_q \gamma_p}{S_\gamma}$	$-\Pi_{qp}^\gamma$	$q \leq F, p > F$
		$q > F, p \leq F$
$e^{-2S} n_q n_p$	$e^{-S} (n_q n_p)^{\frac{1}{2}}$	$q > F, p > F$

$$\gamma_p = n_p h_p + \alpha_p^2 - \alpha_p S_\alpha$$

$$\alpha_p = \begin{cases} e^{-S} h_p, & p \leq F \\ e^{-S} n_p, & p > F \end{cases}$$

$$\Pi_{qp}^\gamma = \left( n_q h_p + \frac{\gamma_q \gamma_p}{S_\gamma} \right)^{\frac{1}{2}} \left( h_q n_p + \frac{\gamma_q \gamma_p}{S_\gamma} \right)^{\frac{1}{2}}$$

$$S = \sum_{q=F+1}^{F+FN_c} n_q, \quad S_\alpha = \sum_{q=F+1}^{F+FN_c} \alpha_q, \quad S_\gamma = \sum_{q=F+1}^{F+FN_c} \gamma_q$$

## 5.3 PNOF7

(PRL 119, 063002, 2017; EPJB 91, 109, 2018)

PNOF7 introduces correlation between pairs of electrons through a term that explicitly contains particle-hole symmetry.

$$E_{el}^{pnof7} = E_{el}^{pnof5} + \sum_{f \neq g} \sum_{p \in \Omega_f} \sum_{q \in \Omega_g}^{N/2} \Pi_{qp}^{\Phi} \mathcal{L}_{pq}$$

$$\Pi_{qp}^{\Phi} = -\Phi_q \Phi_p$$

$$\Phi_q = \sqrt{n_q h_q}$$

$$h_q = 1 - n_q$$

where  $\{n\}$  is the set of natural orbital occupation numbers.

## 5.4 PNOF7s

(PRA 98, 022504, 2018)

In order to choose PNOF7s in DoNOF, you must set IPNOF=7 Ista=1 in the \$NOFINP section of your input file.

In the context of NOF-MP2 (see corresponding section), it is convenient to take into account the inter-pair static correction in the reference-used NOF from the outset, thus preventing the ONs and NOs from suffering an inter-pair non-dynamic influence, however small, in the dynamic correlation domains. This led us to correlate the motion of electrons with parallel and opposite spins belonging to different subspaces as

$$E_{fg} = \sum_{p \in \Omega_f} \sum_{q \in \Omega_g} [n_q n_p (2\mathcal{J}_{pq} - \mathcal{K}_{pq}) - 4\Phi_q^2 \Phi_p^2 \mathcal{K}_{pq}]$$

So the resulting NOF, called static PNOF7 (PNOF7s), can be formulated analogously to PNOF7

$$E_{el}^{pnof7} = E_{el}^{pnof5} + \sum_{f \neq g} \sum_{p \in \Omega_f} \sum_{q \in \Omega_g}^{N/2} \Pi_{qp}^{\Phi} \mathcal{L}_{pq}$$

$$\Pi_{qp}^{\Phi} = -4\Phi_q^2 \Phi_p^2$$

Note real orbitals are assumed, so exchange (K) and time-inversion-exchange (L) integrals are equivalent.



## NOF-MP2

In order to activate the NOF-MP2 calculation in DoNOF, set (in the input file):

```
$NOFINP OIMP2=T IPNOF=7 Ista=1 /
```

Note that static PNOF7 (PNOF7s) is the best option to carry out NOF-MP2 calculations. The latter is specified by the Ista=1 keyword in the input file. Nevertheless, you can use PNOF7 or other functional to generate the reference orbitals.

Current NOFs based on electron pairing take into account most of the non-dynamical effects, and also an important part of the dynamical electron correlation corresponding to the intrapair interactions. Consequently, electron-pairing-based NOFs produce results that are in good agreement with accurate wavefunction-based methods for small systems, where electron correlation effects are almost entirely intrapair. When the number of pairs increases, NOF values deteriorate especially in those regions where dynamic correlation prevails. It is therefore mandatory to add the inter-pair dynamic electron correlation to improve calculations.

The second-order Møller–Plesset (MP2) perturbation theory is the simplest way of properly incorporating dynamic correlation effects. In DoNOF, the recently proposed NOF-MP2 method can be used to obtain a good balance of both static (non-dynamic) and dynamic electron correlation. The zeroth-order Hamiltonian is constructed from a closed-shell-like Fock operator that contains a HF density matrix with doubly and singly occupied orbitals. The electronic energy is then calculated by the expression

$$\begin{aligned}
 E &= \tilde{E}_{hf} + E^{corr} = \tilde{E}_{hf} + E^{sta} + E^{dyn} \\
 \tilde{E}_{hf} &= 2 \sum_{g=1}^{N_\Omega} \mathcal{H}_{gg} + \sum_{f,g=1}^{N_\Omega} (2\mathcal{J}_{fg} - \mathcal{K}_{fg}) - \sum_{g=\frac{N_\Omega}{2}+1}^{N_\Omega} \frac{\mathcal{J}_{gg}}{4} \\
 E^{sta} &= \sum_{g=1}^{N_\Omega/2} \sum_{q \neq p} \sqrt{\Lambda_q \Lambda_p} \Pi_{qp} \mathcal{K}_{pq} - 4 \sum_{f \neq g}^{N_\Omega} \sum_{p \in \Omega_f} \sum_{q \in \Omega_g} \Phi_q^2 \Phi_p^2 \mathcal{K}_{pq} \\
 E^{dyn} &= \sum_{g,f=1}^{N_\Omega} \sum_{p,q > N_\Omega}^{N_B} A_g A_f \langle gf | pq \rangle [2T_{pq}^{gf} - T_{pq}^{fg}] \\
 A_g &= \begin{cases} 1, & 1 \leq g \leq \frac{N_\Omega}{2} \\ \frac{1}{2}, & \frac{N_\Omega}{2} < g \leq N_\Omega \end{cases}
 \end{aligned}$$

For more details about the NOF-MP2 method see Phys. Rev. Lett. 119, 063002 (2017), Phys. Rev. A 98, 022504 (2018), and Phys. Rev. A 100, 032508 (2019).



## INPUT OPTIONS

Use only capital letters in the input file, except for specific cases (see below). For instance, RUNTYP='energy' will not work, but RUNTYP='ENERGY' yes.

### 7.1 Basis Set

The basis set for any calculation can be read from 'basis-name.bas' (see examples in the basis folder of this repository) or introduced in the input file (see example folder of this repository). For historical reasons, we employ the format used in GAMESS US software package.

The file containing the basis sets (basis-name.bas) can be located in the directory where the calculation is performed (\$PWD), in \$HOME/DoNOFsw/basis/, \$HOME/DoNOF/basis/ directories, or you can specify the full path where it is located, that is, PATH/basis-name.bas.

You can find corresponding basis sets for any atomic element in <https://www.basissetexchange.org/>. You must choose the "GAMESS US" format in the "Download basis set" box. In addition, you can use "Advanced" options from the "Download basis set" box to activate different options and versions of the basis set in the "GAMESS US" format, for instance, the "Optimize General Contractions" option.

### 7.2 &INPRUN

The &INPRUN namelist specifies the run type, the atomic basis, and similar fundamental job options. These options are controlled by the following keywords:

RUNTYP: Specifies the run calculation

- = 'ENERGY' single-point energy calculation (Default)
- = 'GRAD' energy + gradients with respect to nuclear coord.
- = 'OPTGEO' optimize the molecular geometry
- = 'HESS' compute numerical hessian from analytic gradients

MULT: Multiplicity of the electronic state

- = 1 singlet (Default)
- = 2,3,... doublet, triplet, and so on

ICHARG: Molecular charge

- = 0 Neutral Molecule (Default)

IECP: Effective Core Potentials

= 0 (Default) All electron calculation

= 1 Read ECP potentials in the \$ECP group

IEMOM: Calculation of electrostatic moments

= 1 calculate dipole moments (Default)

= 2 also calculate quadrupole moments

= 3 also calculate octopole moments

UNITS: Distance units (Any angles must be in degrees)

= ANGS Angstroms (Default)

= BOHR Bohr atomic units

EVEC: An array of the three x,y,z components of the applied electric field, in a.u. (1 a.u. = 1 Hartree/e\*Bohr = 5.1422082(15)d+11 V/m)

= 0.0D0 (Default)

DONTW: Do not write 2e- integrals on the disk (Unit=1)

= T (Default)

ERITYP: Typ of ERIs used in calculations

= FULL 4c ERIs (Default)

= RI 3c/2c ERIs for Resolution of the Identity (RI) App.

= MIX 3c/2c ERIs for Resolution of the Identity (RI) App. once converged change to 4c ERIs (FULL)

GEN: Generative Auxiliary Basis to use in RI Approx. if ERITYP = RI. Values: A2,A2\*,A3,A3\*,A4,A4\*

= A2\* (Default)

SMCD: Symmetric Modified Cholesky Decomposition for the G matrix in the RI Approximation

= F (Default)

HSSCAL: Compute Hessian from analytic gradients and carry out normal mode vibrational analysis at st. point if RUNTYP = OPTGEO (IRUNTYP=3)

= T (Default)

PROJECT: Project Hessian to eliminate rot/vib contaminants

= T (Default)

ISIGMA: Rotational symmetric number for thermochemistry

= 1 There is not a center of symmetry (Default)

= 2 There is a center of symmetry

For more info see <https://cccbdb.nist.gov/thermo.asp>

NATmax: Maximum Number of Atoms

= 100 (Default)

NSHELLmax: Maximum Number of Shells

= 500 (Default)

NPRIMImax: Maximum Number of Gaussian Functions

= 2000 (Default)



## 7.3 &NOFINP

The &NOFINP namelist specifies the type of PNOF calculation, options for the iterative diagonalization method, perturbative corrections, input and output, and similar fundamental job options. These options are controlled by the following keywords:

### 7.3.1 Number of total iterations

MAXIT: Maximum number of OCC-SCF iterations  
= 1000 (DEFAULT)

### 7.3.2 Type of calculation

ICOEF: Energy Optimization with respect to Coefficient Matrix (Natural Orbitals)  
= 0 Optimize only with respect to Gamma variables that determine the occupation numbers  
= 1 Optimize with respect to Gammas and Coefficient matrix (DEFAULT)  
= 2 Optimize only by the orbitals keeping fixed the occupation numbers  
= 3 Optimize by all occupations and core-fragment orbitals. The rest of fragment orbitals remain frozen

IEINI: Calculate only the initial energy  
= 0 (DEFAULT)

NO1: MAX. index of NOs with Occupation equal to 1.0  
= -1 Consider Core NOs (DEFAULT)  
= 0 All NOs are considered  
= Value User specifies how many NOs have OCC equal to 1.0

### 7.3.3 HARTREE-FOCK

RHF: Restricted Hartree-Fock Calculation  
= T (Default)

NCONVRHF: RHF-SCF Density Convergence Criteria  $\text{CONVRHFDM} = 10.0^{**}(-\text{NCONVRHF})$   
= 5 (Default)

MAXITRHF: Maximum number of RHF-SCF iterations  
= 100 (Default)

HFDAMP: Damping of the Fock matrix  
= T (Default)

HFEXTRAP: Extrapolation of the Fock matrix  
= T (Default)

HFID: Use the Iterative Diagonalization Method to generate the HF Orbitals  
= T (DEFAULT)

NTHRESHEID: Convergence of the total energy,  $\text{THRESHEID}=10.0^{**}(-\text{NTHRESHEID})$   
= 6 (DEFAULT)

MAXITID: Maximum number of external iterations  
= 30 (DEFAULT)

KOOPMANS: Calculate IPs using Koopmans' Theorem  
= 0 (DEFAULT)

### 7.3.4 PNOF Selection

IPNOF: Type of Natural Orbital Functional (see section "NOF approximations")  
= 3 PNOF3 + pairing constraints  
= 4 PNOF4 + pairing constraints  
= 5 PNOF5  
= 6 PNOF6  
= 7 PNOF7 (DEFAULT)

Ista: Use Static version of PNOF7  
= 0 PNOF7 (DEFAULT)  
= 1 PNOF7s

HighSpin: Spin-uncompensated calculation type  
= F (DEFAULT) Multiple state ( $M_s=0$ )  
= T High-spin uncompensated state ( $M_s=S$ )

NCWO: Number of coupled weakly occupied MOs per strongly occupied =  $N_c \rightarrow \text{PNOFi}(N_c)$   
= 1 (DEFAULT)  
= 2,3,...  
**=-1 NCWO = NVIR/NDOC where** NVIR: Number of HF virtual MOs ( $\text{OCC}=0$ ), NDOC: Number of strongly occupied MOs

### 7.3.5 Convergence criteria in NOF calculation

For more info see section 3 in Comp. Phys. Comm. 259, 107651 (2021), Code Ocean Capsule; arXiv:2004.06142 [physics.comp-ph] by Piris and Mitxelena

NTHRESHL: Convergence of the Lagrange multipliers,  $\text{THRESHL}=10.0^{**}(-\text{NTHRESHL})$   
= 3 (DEFAULT)

NTHRESHE: Convergence of the total energy,  $\text{THRESHE}=10.0^{**}(-\text{NTHRESHE})$   
= 4 (DEFAULT)

NTHRESHEC: Convergence of the total energy (ORBOPT),  $\text{THRESHEC}=10.0^{**}(-\text{NTHRESHEC})$   
= 10 (DEFAULT)

NTHRESHEN: Convergence of the total energy (OCCOPT),  $\text{THRESHEN}=10.0^{**}(-\text{NTHRESHEN})$

= 10 (DEFAULT)

### 7.3.6 Options for the orbital optimization program (ID method)

For more info and computational details see section 3 in arXiv:004.06142 [physics.chem-ph] by Piris and Mitxelena

MAXLOOP: Maximum Iteration Number for the SCF ITERATION cycle in each ITCALL

= 30 (DEFAULT)

The straightforward iterative scheme fails to converge very often due to the values of some off-diagonal elements  $F_{ki}$ . The latters must be sufficiently small and of the same order of magnitude. A variable factor scales  $F_{ki}$ . We establish an upper bound  $B$ , in such a way that when the absolute value of the matrix element  $F_{ki}$  is greater than  $B$ , it is scaled by a factor  $C_{ki}$  ( $F'_{ki} = C_{ki} * F_{ki}$ ), as to satisfy  $ABS(F_{ki}) \leq B$ .

SCALING: A variable factor scales  $F_{ki}$

= T (DEFAULT)

NZEROS:  $B = 10.0^{**}(1-NZEROS)$ . Initial number of ZEROS in  $F_{ij}$ . The scaling factor varies until the number of ZEROS (.000##) is equal for all elements  $F_{ij}$

= 0 ;  $B = 10.0$  (DEFAULT)

NZEROSm:  $B = 10.0^{**}(1-NZEROSm)$ . Maximum number of zeros in  $F_{ij}$

= 5 ;  $B = 10.0$  (DEFAULT)

NZEROSr:  $B = 10.0^{**}(1-NZEROSr)$ . Number of zeros in  $F_{ij}$  to restart automatically the calculation

= 2 ;  $B = 10.0$  (DEFAULT)

AUTOZEROS: The code select automatically values for NZEROS, NZEROSm & NZEROSr.

Note: Override previously selected values

= T (Default)

ITZITER: Number of Iterations for constant scaling

= 10 (DEFAULT)

DIIS: Direct Inversion in the Iterative Subspace in the orbital optimization if  $DUMEL < THDIIS$  every NDIIS loops

= T (DEFAULT)

NTHDIIS: Energy threshold to begin DIIS

= 3 ;  $THDIIS = 10.0^{**}(-NTHDIIS)$  (DEFAULT)

NDIIS: Number of considered loops to interpolate the generalized Fock matrix in the DIIS

= 5 (DEFAULT)

PERDIIS: Periodic DIIS

= T ; Apply DIIS every NDIIS (DEFAULT)

= F ; DIIS is always applied after NDIIS

### 7.3.7 Options for perturbative calculations

For more info see [PRA 98, 022504 (2018)]

OIMP2: NOF - Orbital Invariant MP2

= F (DEFAULT)

NO1PT2: Frozen MOs in perturbative calculations. Maximum index of NOs with Occupation = 1

= -1 = NO1 (DEFAULT)

= 0 ; All NOs are considered

= Value User specifies how many NOs are frozen

SC2MCPT: SC2-MCPT perturbation theory is used to correct the PNOF5 Energy. Two outputs: PNOF5-SC2-MCPT and PNOF5-PT2

= F (DEFAULT)

NEX: Number of excluded coupled orbitals in the PNOF5-PT2 calculation

= 0 ; All NOs are included (DEFAULT)

### 7.3.8 Restart options for GAMMA, C, diagonal F, and nuclear coordinates

RESTART: Restart from GCF file (DEFAULT=F)

= F ; corresponds to INPUTGAMMA=0,INPUTC=0,INPUTFMIUG=0,INPUTCXYZ=0

= T ; corresponds to INPUTGAMMA=1,INPUTC=1,INPUTFMIUG=1,INPUTCXYZ=1

INPUTGAMMA: Guess for GAMMA variables (determine the Occupation Numbers)

= 0 ; Close Fermi-Dirac Distribution (DEFAULT)

= 1 ; Input from GCF file

INPUTC: Guess for coefficient matrix (Natural Orbitals)

= 0 ; Use HCore or HF Eigenvectors (DEFAULT)

= 1 ; Input from GCF file

INPUTFMIUG: Guess for diagonal elements of the symmetric F matrix (FMIUG0)

= 0 ; Use single diagonalization of Lagrangian (DEFAULT)

= 1 ; Input from GCF file

INPUTCXYZ: Read nuclear coordinates (Cxyz)

= 0 ; From Input file

= 1 ; From GCF file

### 7.3.9 Output options

NPRINT: Output option

- = 0 ; Short Printing (DEFAULT)
- = 1 ; Output at initial and final iterations
- = 2 ; Output everything at each iteration

IAIMPAC: Write information into a WFN file (UNIT 7) for the AIMPAC program

- = 0 ; Not do it
- = 1 ; Do it (DEFAULT)

IFCHK: Write information into Formatted Checkpoint (FCHK) file for visualization software (UNIT 19)

- = 0 ; Don't write
- = 1 ; Write into FCHK file (Default)

MOLDEN: Write information into MLD file for the MOLDEN PROGRAM (UNIT 17)

- = 0 ; Don't write
- = 1 ; Write into MLD file (Default)

NOUSTRDM: Print option for atomic RDMs

- = 0 ; Not do it (DEFAULT)
- = 1 ; Print atomic RDMs in 1DM and 2DM files

NTHRESHDM:  $\text{THRESHDM} = 10.0^{**}(-\text{NTHRESHDM})$

- = 6 (DEFAULT)

NSQT: Print OPTION for 2DM file

- = 0 ; Formatted file
- = 1 ; Unformatted file (DEFAULT)

NOUTCJK: Print option for CJ12 and CK12

- = 0 ; No output (DEFAULT)
- = 1 ; Print CJ12 and CK12 in file 'CJK'

NTHRESHCJK:  $\text{THRESHCJK} = 10.0^{**}(-\text{NTHRESHCJK})$

- = 6 (DEFAULT)

NOUJTijab: Print option for Tijab

- = 0 ; No output (DEFAULT)
- = 1 ; Print Tijab in file 'Tijab'

NTHRESHTijab:  $\text{THRESHTijab} = 10.0^{**}(-\text{NTHRESHTijab})$

- = 6 (DEFAULT)

APSG: Open an APSG file for printing the coefficient matrix (\$VEC-\$END) and the expansion coefficients of the APSG generating wavefunction.

- = F ; No output (DEFAULT)

NTHAPSG: Threshold for APSG expansion coefficients  $\text{THAPSG} = 10.0^{**}(-\text{NTHAPSG})$

= 10 (DEFAULT)

Note: the following options require `NPRINT > 0` to take effect

IWRITEC: Output option for the coefficient matrix

= 0 ; Not do it (DEFAULT)

= 1 ; Do it

IMULPOP: Mulliken population analysis

= 0 ; Not do it (DEFAULT)

= 1 ; Do it

PRINTLAG: Output option for the lagrange multipliers

= F ; Not do it (DEFAULT)

DIAGLAG: Diagonalize Lagrange multipliers. Print new 1e- Energies, Canonical MOs, and new diagonal elements of the 1RDM

= F ; Not do it (DEFAULT)

IEKT: Calculate the Ionization Potentials using the Extended Koopmans' Theorem (EKT)

= 0 ; Not do it (DEFAULT)

= 1 ; Do it

### 7.3.10 Options related to orthonormality of Natural Orbitals

ORTHO: Orthogonalize the initial orbitals

= F ; No

= T ; Yes (DEFAULT)

CHKORTHO: Check the orthonormality of the MOs

= F ; No (DEFAULT)

= T ; Yes

### 7.3.11 Options related to frozen coordinates in geometry optimization

See also "Additional notes" section

FROZEN: Is there any fixed coordinate

= F (DEFAULT)

IFROZEN: By pairs, what coordinate of which atom, e.g. 2,5,1,1 means "y" coordinate of atom 5 and "x" coor of atom 1 to freeze. MAXIMUM of frozen coordinates = 10

= 0 (DEFAULT)

### 7.3.12 Options for optimization program

ICGMETHOD: Define the conjugate gradient method in routines OCCOPTr, CALTijabIsym and OPTIMIZE

- = 1 ; Use SUMSL in CGOCUPSUMSLr, OPTSUMSL, SparseSymLinearSystem\_CG (DEFAULT)
- = 2 ; Use NAG routines E04DGF in OPTCGNAG, CGOCUPNAGr; and F11JEF in SparseSymLinearSystem\_NAG
- = 3 ; Use LBFGS in OPTLBFGS, LBFGSOCUPr

## 7.4 Additional Notes

By default, DoNOF employs the conjugate gradient (CG) method implemented in the “SUMSL” open-source routine to perform the energy optimization with respect to the GAMMA variables (occupation numbers), and the nuclear coordinates if RUNTYP=’OPTGEO’. For more details on SUMSL, see the following references:

J E Dennis, David Gay, and R E Welsch, An Adaptive Nonlinear Least-squares Algorithm, ACM Transactions on Mathematical Software, Volume 7, Number 3, 1981.

J E Dennis, H H W Mei, Two New Unconstrained Optimization Algorithms Which Use Function and Gradient Values, Journal of Optimization Theory and Applications, Volume 28, pages 453-482, 1979.

J E Dennis, Jorge More, Quasi-Newton Methods, Motivation and Theory, SIAM Review, Volume 19, pages 46-89, 1977.

D Goldfarb, Factorized Variable Metric Methods for Unconstrained Optimization, Mathematics of Computation, Volume 30, pages 796-811, 1976.

Alternatively, we have also implemented the LBFGS algorithm written by J. Nocedal (see <http://users.iems.northwestern.edu/~nocedal/lbfgs.html>). This method is activated by setting ICGMETHOD = 3. In our experience, LBFGS works fine for occupation optimization, whereas it must be employed carefully for geometry optimization.

Finally, if you have the NAG library installed, you can use the corresponding subroutines to perform optimizations by setting ICGMETHOD = 2.

### 7.4.1 Geometry Optimization

If RUNTYP=’OPTGEO’ is set, DoNOF automatically sets RHF=F, HFID=F and OIMP2=F at the beginning of the calculation.

It is strongly recommended to set ICGMETHOD=1 (DEFAULT) or ICGMETHOD=2 if you possess the NAG library. In fact, the latter has proven to be much more accurate than LBFGS for this task. The LBFGS algorithm has been employed before in quantum chemistry programs to optimize the geometry (see <http://openmopac.net/Manual/lbfgs.html>). Since LBFGS employs very low memory it is recommended only if a large number of variables is to be optimized. Nevertheless, LBFGS may not work accurately if low-energy interactions are significant in your system.

RUNTYP=’OPTGEO’ may be a computationally demanding task for any ICGMETHOD option. Nevertheless, we have demonstrated (JCP 146, 014102 (2017)) that PNOF approximations produce similar equilibrium geometries for perfect pairing or larger coupling options (i.e. NCWO>1). Therefore, for RUNTYP=’OPTGEO’ is recommended to employ the minimum value of NCWO, that is, run a single-point calculation and check in the output how many weakly-occupied-orbitals have significant occupancies in each subspace. For example, if there are three weakly-occupied-orbitals with non-negligible occupations in each subspace, it will be enough to set NCWO=3 in the RUNTYP=’OPTGEO’ calculation. This can save a large amount of computational time and produce similar equilibrium geometries to those that would be obtained by considering all orbitals correlated with a large basis set.

GCF: All information required to restart any calculation is printed in a file called GCF during the iterative procedure. At the end of the calculation this file is renamed to “name-of-the-molecule.gcf” by our supplied run scripts. It is worth noting that at the end of the GCF the nuclear coordinates are printed. The latter are read at the beginning of the calculation (so the ones from the .inp file are ignored) only if explicitly required by the user, by setting INPUTCXYZ=1 or if RESTART=T in \$NOFINP. This option is particularly useful if the calculation stops unexpectedly during the geometry optimization procedure (RUNTYP='OPTGEO'). If that is the case, run a new calculation setting INPUTCXYZ=1 to converge the energy from the last obtained geometry.

In geometry optimization calculations (RUNTYP='OPTGEO'), you will note that a file named CGGRAD is created during the calculation. Once the calculation ends it is renamed to “name-of-the-molecule.cgo” by our supplied run scripts. This file contains information about the geometry optimization procedure carried out by using the conjugate gradient method (set in the input file by ICGMETHOD), as well as the Hessian and harmonic vibrational frequencies at the solution point. Recall that the Hessian is computed by numerical differentiation of the analytic energy gradients (see details at I. Mitxelena et al. Adv Quant. Chem. ISSN 0065-3276 (2019)), so numerical precision of reported harmonic vibrational frequencies is limited and, apriori, they should be taken only qualitatively.

You may notice in the \$NOFINP section that a keyword FROZEN is used to fix nuclear coordinates during geometry optimization. This is done in cartesians, though it is recommended, for obvious reasons, doing it by using internal coordinates. For the moment this has not been implemented in DoNOF yet. Therefore, we recommend the user to employ FROZEN carefully.

New algorithms and numerical methods for carrying out these optimizations are welcome, so we encourage new collaborations to work on this task.

## 7.4.2 Dependencies

By setting ICGMETHOD=2 in the input file, DoNOF uses the Conjugate Gradient (CG) algorithm coded in NAG library for optimization of the GAMMA variables, as well as nuclear coordinates (if RUNTYP='OPTGEO'). If the user prefers to use NAG subroutines (<https://www.nag.co.uk/content/nag-library>), you must uncomment all lines in the code preceded by '!nag' and link DoNOF code with NAG library. Accordingly, the following routines are called by DoNOF: E04DGF, E04UEF, E04UCF, and F11JEF. The latter is required for perturbative calculations, while the other routines are required for optimization processes.

## 7.4.3 Dissociation

Molecular dissociation is considered the main still unresolved problem of DFT, but of fundamental interest for quantum chemistry. PNOF methods are able to reproduce benchmark potential energy curves of molecular bond dissociation. Nevertheless, this calculation is tricky and must be carried out carefully. In fact, different solutions may arise during the dissociation process depending on the electron correlation present in your system. Computationally it is convenient to converge a single-point calculation, and then start the dissociation process manually by setting: RESTART=F INPUTGAMMA=1 INPUTCOEF=1 INPUTFMIUG=1 ORTHO=T. The restart option allows to use the previous solution, however, we have to avoid reading nuclear geometry from previous point. Since RESTART=T automatically fixes INPUTCXYZ=1, we must employ RESTART=F and specify what we want to read from GCF file, e.g. occupations (INPUTGAMMA=1), orbital coefficients (INPUTCOEF=1), and diagonal elements of pseudofockian (INPUTFMIUG=1).



#### 7.4.4 Symmetry

In DoNOF point-group symmetry is not employed, so C1 symmetry is assumed for any molecular system.

#### 7.4.5 WFN file

The WFN file contains the necessary info to study the output data by using external programs, such as AIMPAC. Note that in this WFN file the energy is referred to as “HF energy”, but it really corresponds to the PNOF energy.

#### 7.4.6 MLD file

The MLD file contains the necessary info to study the output data by using the MOLDEN post processing program of molecular and electronic structure (<https://www3.cmbi.umcn.nl/molden/>)

#### 7.4.7 Numerical Precision

You may notice that different numerical precision is shown for each quantity (orbitals, energy, occupancies, etc) in the output file. The latter is done according to the trustworthy precision inherent to NOF methods. On the contrary, for other purposes is more convenient to employ as much digits as possible.



## EXAMPLES

### 8.1 Single-point calculation

PNOF5 single-point energy calculation of HF molecule with cc-pVDZ basis set starting from RHF calculation:

```
&INPRUN RUNTYP='ENERGY' MULT=1 ICHARG=0 /  
$DATA  
HF-Singlet / Exp. Geom.  
cc-pVDZ  
H 1.0 0.00 0.00 0.0000  
F 9.0 0.00 0.00 0.9168  
$END  
&NOFINP IPNOF=5 RHF=T /
```

PNOF7 single-point energy calculation of the HF molecule at the experimental geometry optimizing only with respect to natural occupation numbers starting from Hartree-Fock calculation:

```
&INPRUN RUNTYP='ENERGY' MULT=1 ICHARG=0 /  
$DATA  
HF-Singlet: cc-pVDZ / Exp. Geom.  
  
H 1.0 0.00 0.00 0.0000  
S 3  
 1 13.0100000 0.0196850  
 2 1.9620000 0.1379770  
 3 0.4446000 0.4781480  
S 1  
 1 0.1220000 1.0000000  
P 1  
 1 0.7270000 1.0000000  
  
F 9.0 0.00 0.00 0.9168  
S 8  
 1 14710.0000000 0.0007210  
 2 2207.0000000 0.0055530  
 3 502.8000000 0.0282670  
 4 142.6000000 0.1064440  
 5 46.4700000 0.2868140  
 6 16.7000000 0.4486410  
 7 6.3560000 0.2647610  
 8 1.3160000 0.0153330  
S 8  
 1 14710.0000000 -0.0001650  
 2 2207.0000000 -0.0013080
```

(continues on next page)

(continued from previous page)

```

3 502.8000000 -0.0064950
4 142.6000000 -0.0266910
5 46.4700000 -0.0736900
6 16.7000000 -0.1707760
7 6.3560000 -0.1123270
8 1.3160000 0.5628140
S 1
1 0.3897000 1.0000000
P 3
1 22.6700000 0.0448780
2 4.9770000 0.2357180
3 1.3470000 0.5085210
P 1
1 0.3471000 1.0000000
D 1
1 1.6400000 1.0000000

$END
&NOFINP IPNOF=7 ICOEF=0 RHF=T /

```

PNOF7 single-point energy calculation of the of Oxygen atom at its triplet state ( $S=1$ ) with STO-3G basis set, optimizing only with respect to natural orbitals:

```

&INPRUN RUNTYP='ENERGY' MULT=3 ICHARG=0 /
$DATA
O-Triplet
STO-3G
O 8.0 0.00 0.00 0.0000
$END
&NOFINP IPNOF=7 ICOEF=2 /

```

## 8.2 Perturbative correction

NOF-MP2 single-point energy calculation of Oxygen atom at its singlet state ( $S=0$ ) with STO-3G basis set:

```

&INPRUN RUNTYP='ENERGY' MULT=1 ICHARG=0 /
$DATA
O-Singlet
STO-3G
O 8.0 0.00 0.00 0.0000
$END
&NOFINP IPNOF=7 Ista=1 OIMP2=T /

```

## 8.3 Geometry Optimization

Geometry Optimization of HF molecule by using cc-pVDZ basis set and NAG conjugate gradient algorithm:

```
&INPRUN RUNTYP='OPTGEO' MULT=1 ICHARG=0 /
$DATA
HF-Singlet: starting from Exp. Geom. ( 0.90148 at HF/CCD level )
cc-pVDZ
H 1.0 0.00 0.00 0.0000
F 9.0 0.00 0.00 0.9168
$END
&NOFINP ICGMETHOD=2 RESTART=T /
```

## 8.4 Convergence

PNOF7 single-point energy + Gradient calculation of Oxygen atom by using STO-3G basis set and convergence criteria of THRESHE=10\*\*(-5) for total energy after both occupation and orbital optimization, THRESHEC=10\*\*(-12) for energy after orbital optimization, and THRESHEN=10\*\*(-16) for energy after occupation optimization. More importantly, set overall convergence of symmetry of matrix Fij as NTHRESHL=4 (usually that is enough, but NTHRESHL=5 is recommended for more accuracy):

```
&INPRUN RUNTYP='GRAD' MULT=1 ICHARG=0 /
$DATA
O-Singlet: STO-3G

O 8.0 0.00 0.00 0.0000
S   3
  1      0.1307093214E+03      0.1543289673E+00
  2      0.2380886605E+02      0.5353281423E+00
  3      0.6443608313E+01      0.4446345422E+00
L   3
  1      0.5033151319E+01     -0.9996722919E-01      0.1559162750E+00
  2      0.1169596125E+01      0.3995128261E+00      0.6076837186E+00
  3      0.3803889600E+00      0.7001154689E+00      0.3919573931E+00
$END
&NOFINP IPNOF=7 AUTOZEROS=F NTHRESHL=4 NTHRESHE=5 NTHRESHEC=12 NTHRESHEN=16 /
```

You can find more examples in the ‘examples’ folder of the DoNOFsw repository (<https://github.com/DoNOF/DoNOFsw/tree/master/examples>)

- [genindex](#)
- [modindex](#)
- [search](#)