Lio

NaN

Departamento de Química Inorgánica, Analítica y Química Física/INQUIMAE, Facultad de Ciencias Exactas y Naturales, Universidad de Buenos Aires, Ciudad Universitaria, Pab. II, Buenos Aires (C1428EHA) Argentina

E-mail:

Abstract

proyecto de manual de lio, hay q ponerle el formato adecuado

1 Quick Reference

This section provides a quick reference for LIO input variables, also providing the default values.

1.1 File I/O

Variable	Type	Default	Description
basis	char*20	'basis'	Filename for the basis set file (if a custom basis set is used).
output	char*20	'output'	Output file.
fCoord	char*20	'qm.xyz'	xyz file (without the header) containing the QM System coordinates.
writexyz	logical	false	Writes an xyz file containing the QM system coordinates, useful when using LIO in tandem with AMBER/GROMACS.
verbose	integer	1	Verbose level.
VCInp	logical	false	
restart_freq	integer	1	
frestart	char*20	'restart.out'	Output restart file.
frestartin	char*20	'restart.in'	Input restart file.

1.2 Electronic Properties

Variable	Type	Default	Description
writeDens	logical	false	Writes electronic density to an output file after calculation.
writeForces	logical	false	Writes final forces to output.
print_coeffs	logical	false	Prints MO coefficients in AO basis.
mulliken	logical	false	Performs a Mulliken Population Analysis.
lowdin	logical	false	Performs a Lowdin Population Analysis.
fukui	logical	false	Calculates condensed-to-atoms Fukui function (Spin Polarized Fukui in open-shell systems).

1.3 Theory Level Options

Variable	Type	Default	Description
natom	integer	0	Number of QM atoms (ignored from AMBER/GROMACS).
nsol	integer	0	Number of classical atoms (ignored from AMBER/GROMACS).
charge	integer	0	QM system total charge.
open	logical	false	Perform an open-shell calculation.
nUnp	integer	0	Number of unpaired electrons.
int_basis	logical	true	If set to false, an external basis file must be provided.
basis_set	char*20	'DZVP'	Name of the basis set used in the calculation. Ignored if int_basis is set to false.
fitting_set	char*40	'DZVP Coulomb Fitting'	Name of the fitting set used in the calculation. Ignored if int_basis is set to false.
nMax	integer	100	Maximum number of SCF steps.
number_restr	integer	0	Amount of distance restraints used.

1.4 Theory Level Advanced Options

Variable	Type	Default	Description
DIIS	logical	true	Use DIIS convergence.
nDIIS	integer	30	
hybrid_converg	logical	false	Use Hybrid convergence.
gold	real*8	10	
told	real*8	1.0D-6	
ETold	real*8	1.0D0	
good_cut	real*8	1.0D-5	
rmax	real*8	16	
rmaxs	real*8	5	
omit_bas	logical	false	
predCoef	logical	false	
dgTrig	integer	100	
iExch	integer	9	
integ	logical	true	
intSolDouble	logical	true	
dens	logical	true	
iGrid	integer	2	
iGrid2	integer	2	

1.5 TD-DFT Options

Variable	Type	Default	Description
timeDep	integer	0	Use TDDFT (when =1).
tdStep	real*8	2.D-5	Timestep for TD-DFT (in atomic units).
ntdStep	integer	0	Maximum number of TD-DFT steps.
propagator	integer	1	
NBCH	integer	10	
field	logical	false	
epsilon	real*8	1.D0	
a0	real*8	1000	
exter	logical	false	Apply an external electric field.
Fx, Fy, Fz	real*8	0.05	The strength of the external electric field.
tdrestart	logical	false	

1.6 Effective Core Potential Options

Variable	Type	Default	Description
ECPMode	logical	false	Activate the ECP mode.
ECPTypes	integer	0	
tipeECP	char*20	'NOT-DEFINED'	
ZListECP	integer	0	
cutECP	logical	true	
ECP_debug	logical	false	
local_nonlocal	integer	0	
ECP_full_range_int	logical	false	
verbose_ECP	integer	0	
fock_ECP_read	logical	false	
fock_ECP_write	logical	false	
fullTimer_ECP	logical	false	
cut2_0	real*8	15.D0	
cut3_0	real*8	12.D0	

1.7 Geometry Optimizations

Variable	Type	Default	Description
steep	logical	false	Activate steepest descend algorithm
Force_cut	real*8	$5 * 10^{-4}$	convergence criteria in forces (Hartree/bohr)
Energy_cut	real*8	$1 * 10^{-4}$	convergence criteria in energy (Hartree)
minimzation_steep	real*8	0.05	initial distance steep (bohr)
n_points	integer	5	number of points scaned for lineal search
n_min_steeps	integer	500	number of optimization steps
lineal_search	logical	true	enable lineal search

1.8 Others

Variable	Type	Default	Description
cubeGen_only	logical	false	
cube_res	integer	40	
cube_sel	integer	0	
cube_dens	logical	false	Prints the electronic density.
cube_dens_file	char*20	'dens.cube'	File containing the electronic density.
cube_orb	logical	false	Prints orbital shapes.
cube_orb_file	char*20	'orb.cube'	File containing the orbital shapes.
cube_elec	logical	false	Prints the electric field.
cube_elec_file	char*20	'field.cube'	File containing the electric field.
cube_sqrt_orb	logical	false	

1.9 GPU Options

Variable	Type	Default	Description
max_function_exponent	integer	10	
little_cube_size	real*8	8.0	
min_points_per_cube	integer	1	
assign_all_functions	logical	false	
sphere_radius	real*8	0.6	
remove_zero_weights	logical	true	
energy_all_iterations	logical	false	
free_global_memory	real*8	0.0	

2 Geometry optimizations

Geometry optimizations or energy minimization is the process of finding an atomic arrangement in space where the force on each atom is acceptably close to zero.

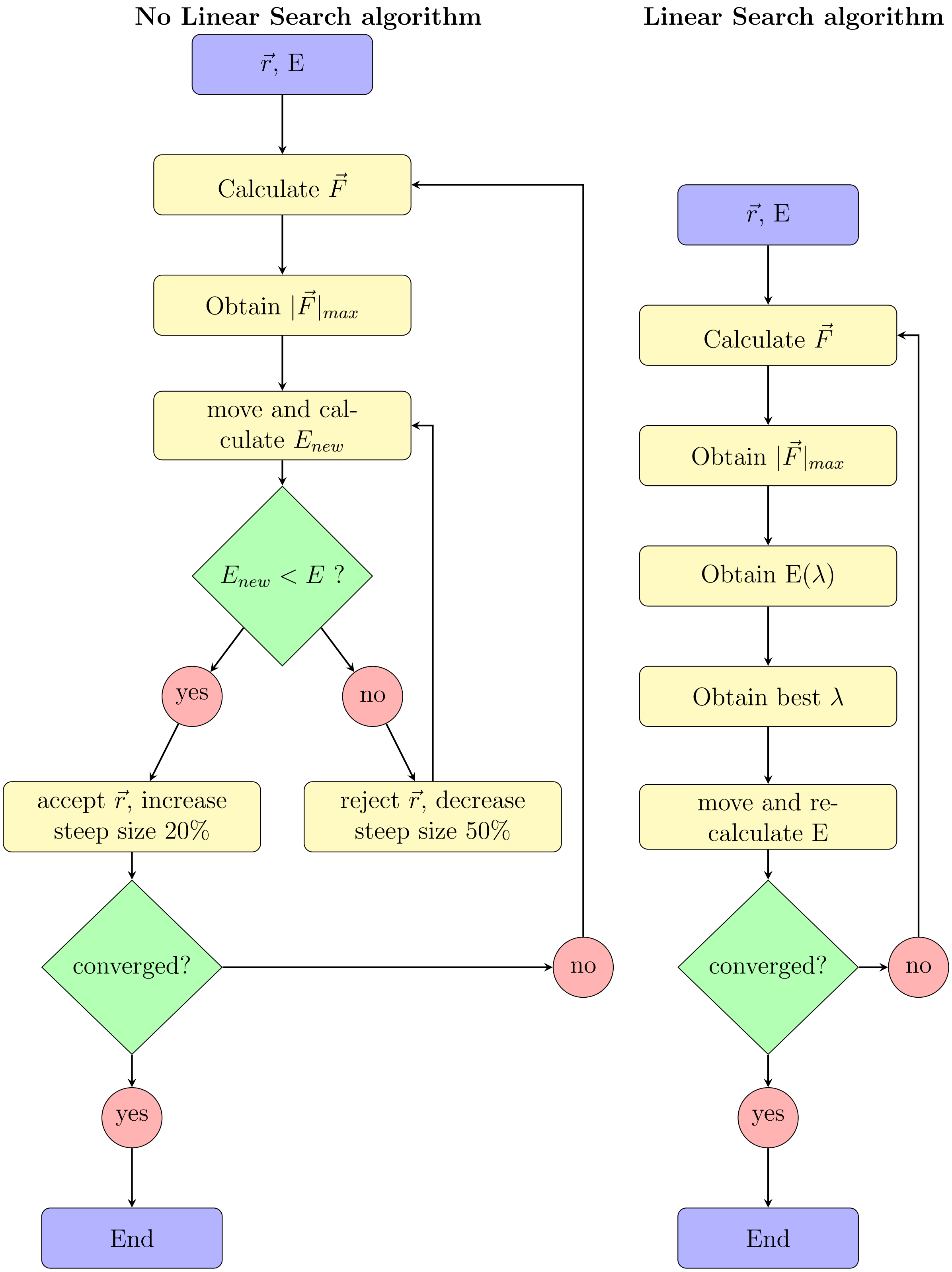
2.1 Implemenation

Lio have a simple steepest-descend algorithm. The idea is just move the system in the force direction a λ value.

$$\vec{r^i}_{new} = \vec{r^i} + \lambda \vec{F}^i \tag{1}$$

In NO lineal search algorithm λ is obtained as $\frac{steep\_size}{|\vec{F}_{max}|}$. If energy decrease with the movement the step is accepted but if energy increase with the steep the steep is rejected and λ is reduced. Each accepted movement increase steep size a 20% and each rejected move decrease steep size a 50%.

In lineal search algorithm the system scan energy as function of λ and predict the best value of λ for move system in force direction.


No Linear Search algorithm

Linear Search algorithm

Best λ in lineal search algorithm is obtained by a quadratic function ajust using minimun energy of the scan and previus and next points.

2.2 Using geometry optimizations

Adding steep=t in lio input enables geometry optimization (steepest descend, lineal search by default). Cutoff criteria are sets by Force_cut and Energy_cut (5E-4 Hartree/bohr and 1E-4 Hartree by Default). Number of minimization steeps is set by n_min_steeps (500 by default) and initial distance steep is set by minimzation_steep (by default 0.05 bohr)
It is recomendable to compile lio in double precision for minimice error in exchange-correlation forces (full_double=1).

Outputs of geometry optimizations are traj.xyz (atoms coordinates in each steepes descend movement) and optimization.out (steep, energy and others). If verbose=true optimization.out include energy of each lineal search point.

2.3 Examples

Examples of geometry optimization are made in lio/test/13_geom_optim.

3 Restraints

LIO may add an extra potential term to the Hamiltonian in order to restrain the distance between specified pairs of atoms.

3.1 Implemenation

The implementation is a simple harmonic potential over a generalized coordinate r.

$$U = \frac{1}{2}k[r - l_0]^2 \tag{2}$$

r may be defined as a weighted combination of distances between pairs of atoms.

$$r = \sum_i \sum_{j>i} w_{ij} |\vec{r_i} - \vec{r_j}| \tag{3}$$

In this formulation the force over an atom l is:

$$\vec{F_l} = -k[r - l_0] \sum_i \sum_{j>i} w_{ij} \frac{\vec{r_{ij}}}{r_{ij}} \eta_{ijl} \tag{4}$$

Where η_{ijl} is defined as:

$$\eta_{ijl} = \begin{cases} 1 & \text{if } l = i \\ -1 & \text{if } l = j \\ 0 & \text{in other case} \end{cases}$$

3.2 Using Restraints

The number of pairs of atoms to be added in the restraint potential(s) is defined by setting the variable number_restr, and a list of distance restrains have to be added to in an additional lio.restrain file. For example:

a_i	a_j	index	k	w_{ij}	l_0
1	2	0	0.1	1.0	7.86
3	4	0	0.1	-1.0	7.86
7	9	1	0.4	2.0	-2.3
13	1	1	0.4	1.0	-2.3
14	3	1	0.4	-3.0	-2.3
14	2	2	0.2	1.0	0.5
8	5	3	0.3	1.0	3.2

Columns a_i and a_j contain the atom numbers in the QM system to be restrained, while the index number determines which distances contribute to a same generalized reaction coordinate. The remaining columns are the force constants (k), weights of that distance in the generalized coordinate (w_{ij}) and equilibrium positions in atomic units (l_0).

3.3 Examples

1)In lio.in:

number_restr = 1

in lio.restrain:

a_i	a_j	index	k	w_{ij}	l_0
1	2	0	0.1	1.0	7.86

Potential added to system:

$$U = \frac{1}{2}0.1\Big[1.0|\vec{r_1} - \vec{r_2}| - 7.86\Big]^2 \tag{5}$$

2)In lio.in:

number_restr = 2

in lio.restrain:

a_i	a_j	index	k	w_{ij}	l_0
1	2	0	0.1	1.0	7.86
3	4	0	0.1	-1.0	7.86

Potential added to system:

$$U = \frac{1}{2}0.1\Big[1.0|\vec{r_1} - \vec{r_2}| - 1.0|\vec{r_3} - \vec{r_4}| - 7.86\Big]^2 \quad (6)$$

3)In lio.in:

number_restr = 4

in lio.restrain:

a_i	a_j	index	k	w_{ij}	l_0
1	2	0	0.1	1.0	7.86
3	4	0	0.1	-1.0	7.86
1	3	1	0.3	3.5	-2.31
7	8	1	0.3	-2.2	-2.31

Potential added to system:

$$U = \frac{1}{2}0.1\Big[1.0|\vec{r_1} - \vec{r_2}| - 1.0|\vec{r_3} - \vec{r_4}| - 7.86\Big]^2 + \frac{1}{2}0.3\Big[3.5|\vec{r_1} - \vec{r_3}| - 2.2|\vec{r_7} - \vec{r_8}| + 2.31\Big]^2 \quad (7)$$