Guide for use of the modified APE code

H. Francisco and S.B. Trickey
Quantum Theory Project,
Dept. of Physics, University of Florida
Gainesville FL 32611[*]

(Dated: 20 July 2021; version 4)

I. INTRODUCTION

The Atomic Pseudo-potentials Engine (APE) code is a computer package designed to generate and test norm-conserving pseudo-potentials within Density Functional Theory (DFT). It was developed by M.J.T. Oliveira and F. Nogueira [1]. The APE code can be obtained from `http://www.tddft.org/programs/APE/`. It is licensed under GPLv2. We gratefully acknowledge that licensure.

The pseudo-potentials generated by APBE can be non-relativistic, scalar relativistic, or fully relativistic. They can include semi-core states explicitly.

In 2018, K. Luo modified version 2.2.1 of APE in order to do orbital-free DFT (OF-DFT) calculations. This Guide documents those modifications. Documentation on the use of the original code is provided from the APE web site and in the download. Note that the modified code falls under the same licensure as the original APE code.

The implementation is based upon the OF-DFT modification of the Kohn-Sham (KS) equation [2, 3] that follows from decomposition of the KS kinetic energy functional into Von Weizsäcker and Pauli terms [4]. The result is

$$\left[-\frac{1}{2}\nabla^2 + v_\theta + v_{eff}^{KS}\right] n^{1/2}(\mathbf{r}) = \mu n^{1/2}(\mathbf{r}) \tag{1}$$

with

$$v_\theta = \frac{\delta T_\theta[n(\mathbf{r})]}{\delta n(\mathbf{r})}, \tag{2}$$

and

$$v_{eff}^{KS} = \frac{\delta(E_{ext}[n(\mathbf{r}]) + E_H[n(\mathbf{r})] + E_{xc}[n(\mathbf{r})])}{\delta n(\mathbf{r})} \,. \tag{3}$$

In these, v_θ is the Pauli potential which is the functional derivative of the Pauli KE $T_\theta[n(\mathbf{r})]$. $E_{ext}[n(\mathbf{r}]$ is the external field interaction energy, $E_H[n(\mathbf{r})]$ the Hartree energy (classical electron-electron repulsion), and $E_{xc}[n(\mathbf{r})]$ the exchange-correlation (XC) energy functional.

In principle, Eq. (1) can be solved by simple modification of the self-consistent loop in a conventional KS code such as APE. In practice, that strategy has limitations. See Ref. 4 and references therein. Despite those limitations, it is worthwhile to have the modified APE code as a tool for OF-DFT development and use.

[*] francisco.hector@ufl.edu; trickey@qtp.ufl.edu

II. INSTALLATION INSTRUCTIONS

A. prerequisites

- Intel compilers (currently APE only works with Intel compilers)

- MPI libraries (seem to be required) We used the Intel® MPI Librarys (`https://software.intel.com/content/www/us/en/develop/tools/oneapi/hpc-toolkit/download.html?operatingsystem=linux&distributions=yumpackagemanager/`)

- GSL=GNU Software Library [5] (version 2.3 on Debian 9 "stretch" and version 2.6-4 on Fedora 34 GNU/LINUX, both tested)

- Libxc, version 2.0.0 or higher (version 2.0.0 and 2.1.2, both tested)
  Libxc can be obtain from `https://www.tddft.org/programs/libxc/download/previous/`

B. Download and installation of libraries

Before installing APE itself, first install all of the prerequisites. If libxc is not already installed, download, extract and install it as follows (assuming the download directory is ~/Downloads)

```
 cd ~/Downloads/
tar -xvf libxc-2.0.0.tar.gz
cd libxc-2.0.0
export libxc_dir=$(pwd) (one can move and rename the installation directory as one wishes)
export CC=icc
export FC=ifort
./configure –prefix=$libxc_dir
make -j4
make install
```

Ready, libxc is installed already.

The modified version of APE can be downloaded from `http://www.qtp.ufl.edu/ofdft/research/computation.shtml`. After downloading the tarball, proceed as follows.

```
cd ~/Downloads/
tar -xvf ape-2.2.1.tar.xz
cd ape-2.2.1
export APE_dir=$(pwd) (one can move and rename the installation directory as one wishes)
export CC=icc
export FC=ifort
export FCCPP=fpp
```

./configure –with-libxc-prefix=$libxc_dir –with-gsl-prefix=/usr –prefix=$APE_dir (here one can note that GSL library is used during APE installation)

make -j4

make install

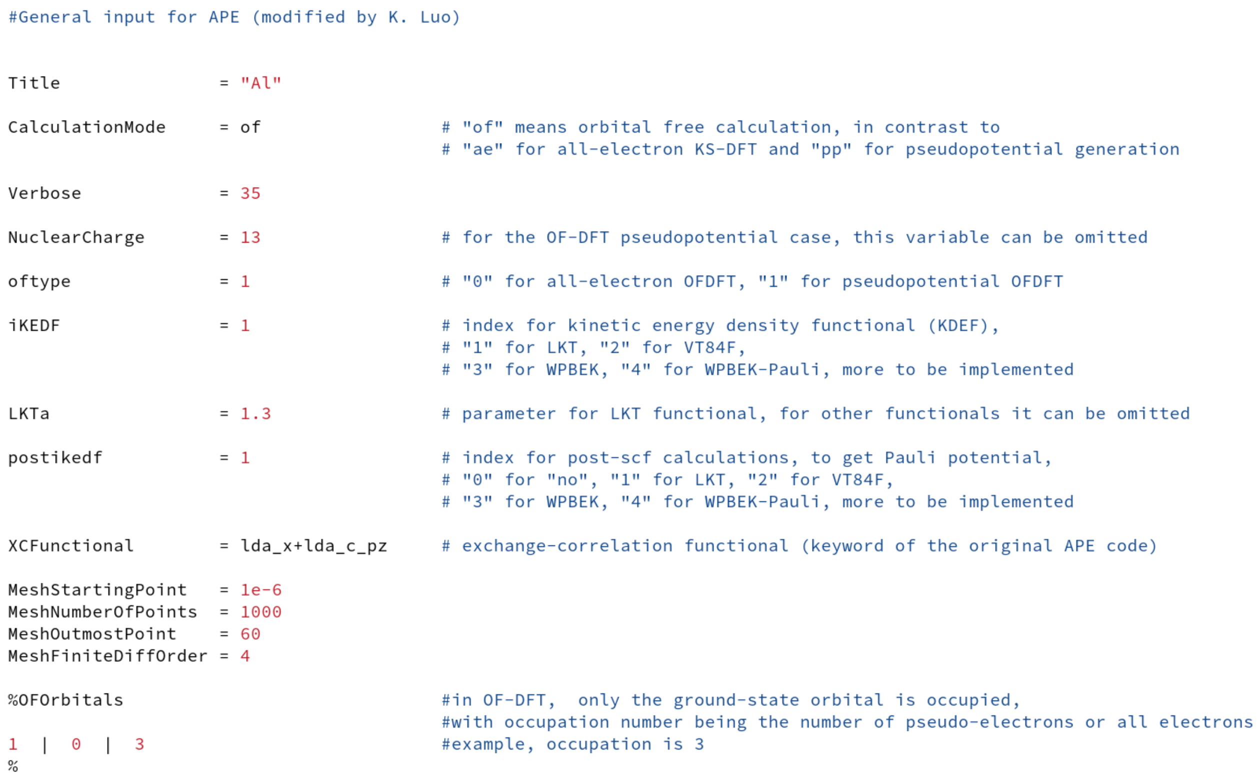
After “make” is finished, there is an executable called “ape” in $APE_dir/src/. That executable is for running the OF-DFT or conventional KS-DFT calculations.

III. FIRST STEPS AND EXAMPLES

Fig. 1 displays the general input file to peform OF-DFT calculations. Variables not in the original APE code were introduced by Luo. These variables are case-insensitive.

To run an APE job, use

$APE_dir/src/ape < input < output



```
#General input for APE (modified by K. Luo)


Title                = "Al"

CalculationMode      = of                # "of" means orbital free calculation, in contrast to
                                         # "ae" for all-electron KS-DFT and "pp" for pseudopotential generation

Verbose              = 35

NuclearCharge        = 13                # for the OF-DFT pseudopotential case, this variable can be omitted

oftype               = 1                 # "0" for all-electron OFDFT, "1" for pseudopotential OFDFT

iKEDF                = 1                 # index for kinetic energy density functional (KDEF),
                                         # "1" for LKT, "2" for VT84F,
                                         # "3" for WPBEK, "4" for WPBEK-Pauli, more to be implemented

LKTa                 = 1.3               # parameter for LKT functional, for other functionals it can be omitted

postikedf            = 1                 # index for post-scf calculations, to get Pauli potential,
                                         # "0" for "no", "1" for LKT, "2" for VT84F,
                                         # "3" for WPBEK, "4" for WPBEK-Pauli, more to be implemented

XCFunctional         = lda_x+lda_c_pz    # exchange-correlation functional (keyword of the original APE code)

MeshStartingPoint    = 1e-6
MeshNumberOfPoints   = 1000
MeshOutmostPoint     = 60
MeshFiniteDiffOrder  = 4

%OFOrbitals                              #in OF-DFT,  only the ground-state orbital is occupied,
                                         #with occupation number being the number of pseudo-electrons or all electrons
1  |  0  |  3                            #example, occupation is 3
%
```

FIG. 1. General input for the APE code with description of keywords. Note: for post-SCF calculations, omit the keyword “iKEDF”.

We give two examples of APE usage. They reproduce the calculations for the Al pseudo-atom in the $3s^2\ 3p^1$ configuration presented in the LKT paper [6].

The first example is an OF-DFT calculation using an atom-centered local pseudo-potential from bulk-derived valence electron densities (BLPS) [7, 8] with the Perdew-Zunger (PZ) XC local density approximation (LDA) [9].

The second example is a post-SCF calculation that uses the self-consistent solution of the ordinary KS equation with the same pseudo-potential and the same XC approximation mentioned before. It generates a pseudo-density and from it obtains the Pauli kinetic energy density (KED), t_θ, and the ingredients to construct the Pauli Potential, v_θ (see Eq. 25 in ref [10]) of that Pauli KEDF.

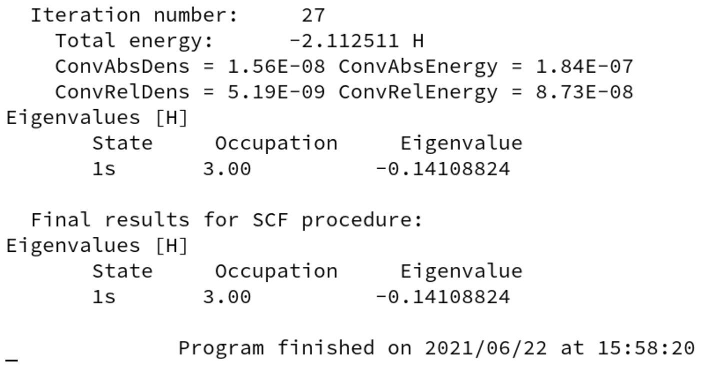
APE uses one input file to make any calculation. However, for calculations with pseudo-potentials, inside the directory for the calculation, the file “lpp” must be present and linked to “full.lps”. It contains the local pseudopotential for the target, in the case of these examples, the Al atom. The BLPS pseudo-potential can be obtained from `https://github.com/PrincetonUniversity/BLPSLibrary/`). (Note that K. Luo *et al.* extended the tail of the potential up to 60 Bohr.)

The first example is in $APE_dir/sample-ofdft/examples/Al-LKT/OF-DFT/

The second example is in $APE_dir/sample-ofdft/examples/Al-LKT/post-SCF/ (Important note: for post-SCF calculations, the keyword “iKEDF” must be omitted.)

In both cases, the calculation will generate an output directory called “of”. In that directory one can find the following files:

- **ouput file**. This file shows what is written when the calculation finishes completely and correctly.



```
  Iteration number:     27
    Total energy:      -2.112511 H
    ConvAbsDens = 1.56E-08 ConvAbsEnergy = 1.84E-07
    ConvRelDens = 5.19E-09 ConvRelEnergy = 8.73E-08
Eigenvalues [H]
      State     Occupation     Eigenvalue
      1s       3.00          -0.14108824

  Final results for SCF procedure:
Eigenvalues [H]
      State     Occupation     Eigenvalue
      1s       3.00          -0.14108824

             Program finished on 2021/06/22 at 15:58:20
_
```

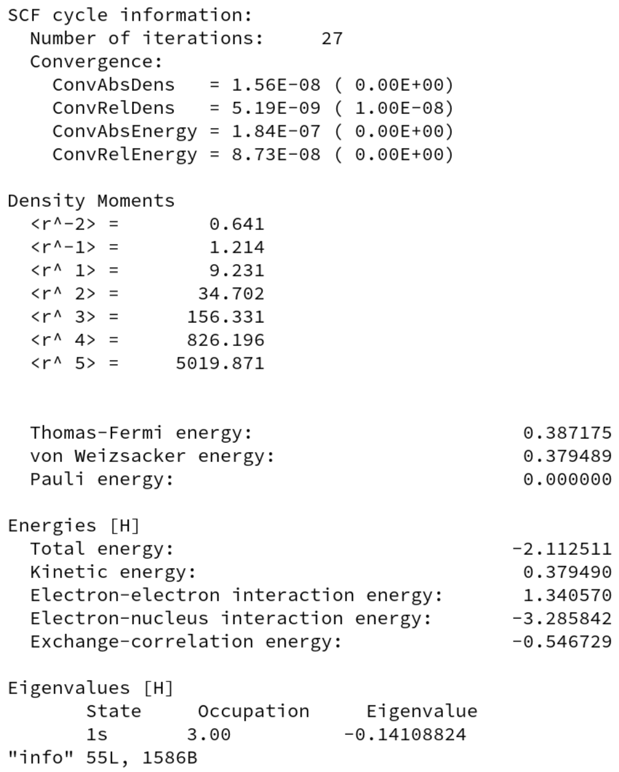
FIG. 2. Example of the output file.

- **info**. This file contains the summary from the calculation, as well as the value of the system total energy and OFDFT eigenvalues, see Fig. 3.

- **locpot.dat**. This file contains the data for plotting the BLPS local pseudo-potential as a function of radial position.

- **postscf_vtheta**. This file contains the data for plotting v_θ and t_θ/n as functions of radial position, when one does OF-DFT or post-SCF calculations using a Pauli functional.

- **vtheta**. This file contains the data for plotting the Pauli potential v_θ as a function of radial position for conventional KS calculations.



```
SCF cycle information:
  Number of iterations:     27
  Convergence:
    ConvAbsDens   = 1.56E-08 ( 0.00E+00)
    ConvRelDens   = 5.19E-09 ( 1.00E-08)
    ConvAbsEnergy = 1.84E-07 ( 0.00E+00)
    ConvRelEnergy = 8.73E-08 ( 0.00E+00)

Density Moments
  <r^-2> =        0.641
  <r^-1> =        1.214
  <r^ 1> =        9.231
  <r^ 2> =       34.702
  <r^ 3> =      156.331
  <r^ 4> =      826.196
  <r^ 5> =     5019.871


  Thomas-Fermi energy:                          0.387175
  von Weizsacker energy:                        0.379489
  Pauli energy:                                 0.000000

Energies [H]
  Total energy:                                -2.112511
  Kinetic energy:                               0.379490
  Electron-electron interaction energy:         1.340570
  Electron-nucleus interaction energy:         -3.285842
  Exchange-correlation energy:                 -0.546729

Eigenvalues [H]
       State     Occupation     Eigenvalue
       1s       3.00          -0.14108824
"info" 55L, 1586B
```

FIG. 3. Example of the info file.

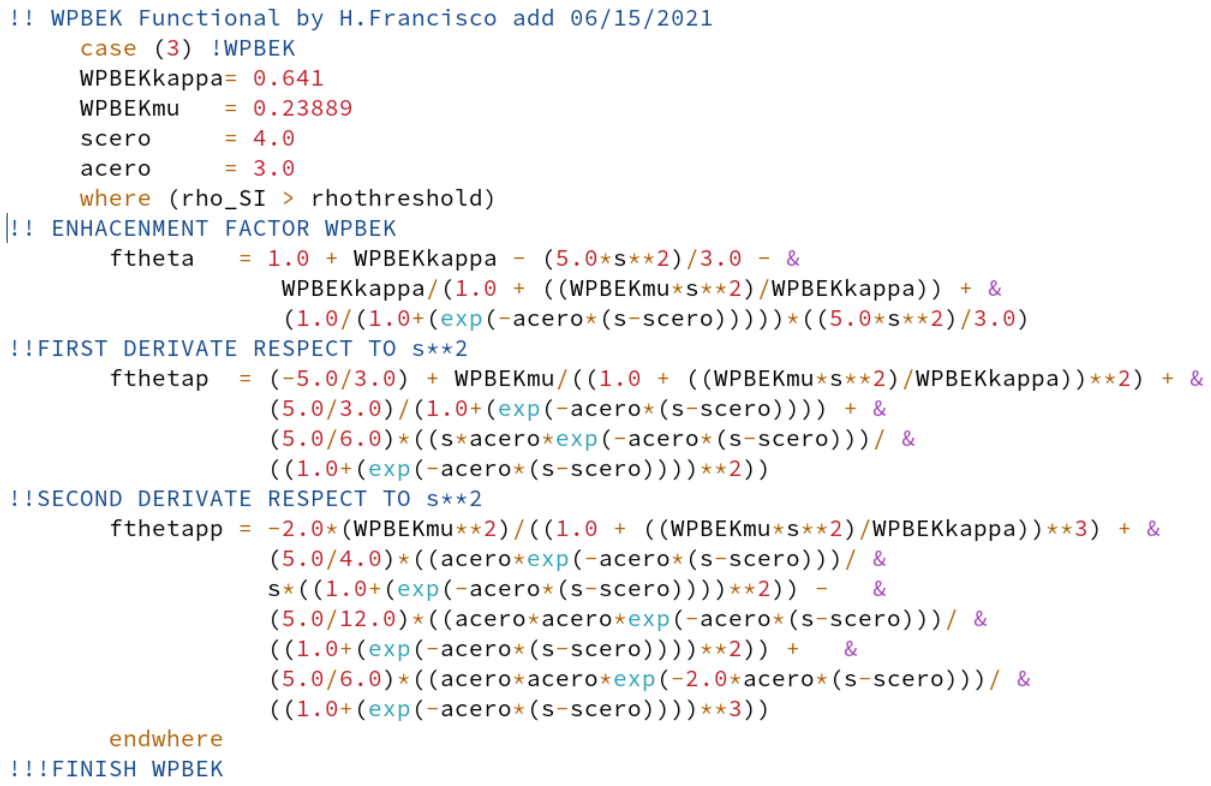
- **t_theta**. This file contains the data for potting t_θ/n as a function of radial position for conventional KS calculations.

- **density**. This file contains the data for plotting the pseudo-density as a function of radial position if one does a post-SCF calculation. Otherwise, it contains the data for plotting the pseudo-density generated by a Pauli functional in an OF-DFT calculation as a function of radial position.

IV. IMPLEMENTING A PAULI KINETIC ENERGY FUNCTIONAL

To implement a new Pauli kinetic energy functional of generalized-gradient approximation (GGA) type in the modified APE code, it is necessary to modify the "states_batch.F90" Fortran file located in the directory $APE_dir/ape-2.2.1/src/. In that file, locate the subroutine called "pauli_potential". In it, one must add a new case for each new Pauli functional. For example, Fig. 4 displays the implementation of the WPBEK functional [11]. One can see that "ftheta" is the Pauli enhancement factor, "fthetap" is the first derivative of the Pauli enhancement factor with respect to s^2, and "fthetapp" is the second derivative with respect to s^2. Here, s is the familiar reduced density gradient

$$s(\mathbf{r}) := \frac{|\nabla n(\mathbf{r})|}{(3\pi^2)^{1/3} n^{4/3}(\mathbf{r})} \ . \tag{4}$$



```
!! WPBEK Functional by H.Francisco add 06/15/2021
     case (3) !WPBEK
     WPBEKkappa= 0.641
     WPBEKmu    = 0.23889
     scero      = 4.0
     acero      = 3.0
     where (rho_SI > rhothreshold)
!! ENHACEMENT FACTOR WPBEK
       ftheta   = 1.0 + WPBEKkappa - (5.0*s**2)/3.0 - &
                   WPBEKkappa/(1.0 + ((WPBEKmu*s**2)/WPBEKkappa)) + &
                   (1.0/(1.0+(exp(-acero*(s-scero)))))*((5.0*s**2)/3.0)
!!FIRST DERIVATE RESPECT TO s**2
       fthetap  = (-5.0/3.0) + WPBEKmu/((1.0 + ((WPBEKmu*s**2)/WPBEKkappa))**2) + &
                  (5.0/3.0)/(1.0+(exp(-acero*(s-scero)))) + &
                  (5.0/6.0)*((s*acero*exp(-acero*(s-scero)))/ &
                  ((1.0+(exp(-acero*(s-scero))))**2))
!!SECOND DERIVATE RESPECT TO s**2
       fthetapp = -2.0*(WPBEKmu**2)/((1.0 + ((WPBEKmu*s**2)/WPBEKkappa))**3) + &
                  (5.0/4.0)*((acero*exp(-acero*(s-scero)))/ &
                  s*((1.0+(exp(-acero*(s-scero))))**2)) -   &
                  (5.0/12.0)*((acero*acero*exp(-acero*(s-scero)))/ &
                  ((1.0+(exp(-acero*(s-scero))))**2)) +   &
                  (5.0/6.0)*((acero*acero*exp(-2.0*acero*(s-scero)))/ &
                  ((1.0+(exp(-acero*(s-scero))))**3))
       endwhere
!!!FINISH WPBEK
```

FIG. 4. Implementation of WPBEK functional in subroutine “pauli_potential”.

The case number is related to the “iKEDF” and “postikedf” keywords of the input file. Currently there are four OFDFT cases,

case (1), LKT functional [6].

case (2), VT84F functional [12].

case (3), WPBEK functional [11].

case (4), WPBEK-Pauli functional [11].

Finally, K. Luo mentioned these limitations for the modified APE code.

- More GGA functionals could be implemented easily but testing requires more effort. The difficulty is that the single-orbital equation based on the scf loop is not guaranteed to converge for an arbitrary KEDF. See Ref. 4 and references therein.

- It seems more difficult to run the all-electron orbital-free calculation than the OF-DFT LPP calculation. The all-electron Li atom was tested by Luo.

Acknowledgments The original versions of several parts of this document were drafted by Kai Luo in 2018. We are grateful as well both for his modifications of APE and for his recent assistance in restoring the modified APE code to active status. This work was supported by U.S. National Science Foundation grant DMR-1912618.

[1] M.J.T. Oliveira and F. Nogueira, Comput. Phys. Commun. **178**, 524 (2008).
[2] M. Levy, J.P. Perdew, V. Sahni, Phys. Rev. A **30** 2745 (1984).
[3] M. Levy and H. Ou-Yang, Hui, Phys. Rev. A **38**, 625 (1988).
[4] V.V. Karasiev and S.B. Trickey, Comput. Phys. Commun. **183**, 2519-2527 (2012).
[5] GSL is obtainable from `http://www.gnu.org/software/gsl`.
[6] K. Luo, V.V. Karasiev, and S.B. Trickey, Phys. Rev. B **98**, 041111 (2018).
[7] B. Zhou, Y.A. Wang, and E.A. Carter, Phys. Rev. B **69**, 125109 (2004).
[8] C. Huang and E. A. Carter, Phys. Chem. Chem. Phys. **10**, 7109 (2008).
[9] J.P. Perdew, and A. Zunger, Phys. Rev. B **23**, 5048 (1981).
[10] V.V. Karasiev, R.S. Jones, S.B. Trickey, and F. E. Harris, Phys. Rev. B **80**, 245220 (2009).
[11] H.I. Francisco, J. Carmona-Espíndola, and J.L. Gázquez, J. Chem. Phys. **154**, 084107 (2021).
[12] V.V. Karasiev, D. Chakraborty, O.A. Shukruto, and S.B. Trickey, Phys. Rev. B **88**, 161108 (2013).