tise_1D_pseudo User's Guide

Curro Perez-Bernal <francisco.perez@dfaie.uhu.es>
Laura Moschini <laura.moschini@pd.infn.it>

Version 1

Contents

Chapter 1

Introduction

1.1 What's `tise_1D_pseudo`?

The `Fortran 90` programs and `Perl` scripts in the `tise_1D_pseudo` suite solve the one dimensional time-independent Schroedinger equation for a one body problem and for several potentials using a expansion in three possible bases: harmonic oscillator (HO), transformed harmonic oscillator (THO), and infinite square well box (BOX). The full spectrum and eigenstates are computed. This includes both the (negative energy) bound states and the positive energy pseudostates that discretize the continuum part of the spectrum (positive energies).

1.2 Copyright statement

Chapter 2

Installation guide

The libraries lapack95, minpack, and intlib libraries are needed. The program files can be found in `git` and a zip file can be dowloaded using `wget` as follows

```
wget https://github.com/currix/tise_1D_Pseudo/archive/master.zip
```

The files should be unpacked and the usual compilation steps follow

```
$ unzip master.zip
Archive:  master.zip
3309629b189bc4a2ca777862fd102d7df45276bf
   creating: tise_1D_Pseudo-master/
 extracting: tise_1D_Pseudo-master/.gitignore
  inflating: tise_1D_Pseudo-master/LICENSE
       .
       .
       .
  inflating: tise_1D_Pseudo-master/test/tho_wsaxon_1D_1body.inp
$ ls
master.zip  tise_1D_Pseudo-master/
$ cd tise_1D_Pseudo-master/
$ ls
bin/  LICENSE  README  REFERENCES  src/  test/
$ cd src/
$ make clean
[pts/4][curro.deckard: src]$ make all
gfortran -c  -Wall -I/usr/local/lib/lapack95/lapack95_modules -I/usr/local/include/lapack95_modules -c nrtype.f90
     .
     .
     .
gfortran -o ../bin//morse_1body_1D_ISQW  nrtype.o constants.o pot_param.o egs_mod_isqw.o infsq_box_1D.o build_ISQW_bas.o ham_mat_1D_1
```

The directory structure is the following

1. `bin`: executable files.
2. `src`: `Fortran 90` source files and compilation `Makefile`.
3. `doc`: program documentation in several formats.
4. `test`: input files

Once compiled, the most of the resulting program names follow the convention *potential*`_1body_1D_`*BAS* where the potentials included are *potential*

1. Woods Saxon (`wsaxon`)
2. Woods Saxon with a barrier (`barrierws`)
3. Mexican hat (`mexhat`)
4. Ginocchio (`ginocchio`)
5. Morse (`morse`)
6. Poeschl-Teller (`pteller`)

7 Hazi and Taylor (`ht`).

The available bases for the Hamiltonian construction and diagonalization are

1 Harmonic Oscillator (HO)

2 Transformed Harmonic Oscillator (THO)

3 Infinite Square Well Box (ISQW)

Other executable files:

1 `program_launcher_1body_N`: `Perl` script to launch the provided programs to look for energy convergence or to prepare grace figure files.

2 `dBde_script`: `Perl` script to compute dB/dE from the B1 or B2 program output for pseudostates.

If the programs works correctly in your system you can also, running as *root* user, install them, by default under the directory `/usr/local/bin`. The associated Perl scripts will also be installed in this directory. In this case it is very convenient to add the directory`/usr/local/bin` to the default `$PATH` variable. As aforementioned, in order to install the program it you should do it as superuser or using `sudo`

```
$ sudo make install
[sudo] password for sudouser:
$ ls  /usr/local/bin/
```

If you want to remove these files you can also do it executing `sudo make uninstall`.

Chapter 3

Getting Started

We proceed to describe the input file and the program execution for each case.

3.1 Input file for the Harmonic Oscillator (HO) basis case

The provided input file is a `namelist` input file with the following structure (e.g. `test/harm_osc_1D_1body_wsaxon.inp`

```
#
#       INPUT FILE FO HO_1BODY_1D
#
&INP_X     X_MIN = -140.0, X_MAX = 140.0 /
&INP_DIM   DIM_X = 1501, DIM_HO = 500, last_bound_state = 3, max_aval_har = 10.0 /
&INP_MASS  IAD = 4, REDUCED_MASS = 0.975 /
&INP_POT   PARAM_POT = -50.0, 2.0, 0.4, 0.0, 0.0  /
&INP_SHIFT I_PHASE = 0, LAMBDA = 0.1 /
#
&INP_AUX   I_gs = 0, Isave_EN = 1, Isave_WF = 0, Isave_BAS = 0, I_sumr = 0, I_toten = 3, B_analytical = .T., B_numerical = .F, Iprint
#
#
```

Variables defined:

1 `X_min, X_max`: *float* type. Spatial interval where the harmonic oscillator basis is defined expressed in *fm* units.

2 `Dim_X`: *integer* type. Number of points in the spatial grid from `X_min` to `X_max`.

3 `Dim_HO`: *integer* type. Size of the truncated HO basis used in the calculation.

4 `Last_Bound_State`: *integer* type. If the potential has several bound states, state used to estimate the HO length scale.

5 `Max_aval_har`: *float* type. If different from zero, calculate only system eigenstates with energies less than `Max_aval_har` (MeV units). If `Max_aval_har = 0.0` the program computes all the eigenstates.

6 `IAD`: *integer* type. If `IAD = 1` then unitless quantities are defined (hbar^2/1 amu = 1) and `kmin = 1` else we optimize the `kmin` value to minimize the energy of the ground state using a N=1 basis. (To Do: recipe for weakly bound states).

7 `REDUCED_MASS`: *float* type. Reduced mass of the system expressed in *amu*.

8 `PARAM_POT`: *float* type. Potential parameters.

9 `I_PHASE`: *integer* type. If this variable is not zero the phase shifts are computed using the Hazi and Taylor approach.

10 `Lambda`: *float* type. Lambda parameter in the Hazi and Taylor calculation of the phase shift[1].

11 `I_gs`: *integer* type. If `I_gs = 1` the `gs_wavefunction.dat` file is saved with the system ground state wavefunction and a header with the system info.

[1] See A. U. Hazi and H. S. Taylor. Phys. Rev. A, *1*:1109â1120, 1970.

12 `Isave_EN`: *integer* type. If `Isave_EN = 1` the following data are saved. The system eigenvalues are saved in a file with name `ho_eigenvalues_N`*`dimbasis`*`.dat` file is saved (unit 73) with the system eigenvalues and a header with the system info. The system potential and eigenstates are saved in a file with name `ho_poteigvec_N`*`dimbasis`*`.dat` file (unit 74) with the following format

```
x  Potential(x) 10xEigvec_1(x)+Eigval(1) 10xEigvec_2(x)+Eigval(2) .... 10xEigvec_dim(x)+Eigval(2)
```

Another file, named `ho_pot_eigvec2_N`*`dimbasis`*`.dat` file is saved (unit 75) with the same format than the previous one but storing the squared values of the wavefunctions as a function of the spatial coordinate.

13 `Isave_WF`: *integer* type. If `Isave_WF = 1` the eigenvectors and eigenvector derivatives are saved in files named `ho_eigenvectors_N`*`dimbasis`*`.dat` and `ho_eigvec_der_N`*`dimbasis`*`.dat` with the format

```
x  Eigvec_1(x) Eigvec_2(x) ... Eigvec_dim(x)
```

The units involved are unit 71 and 72, respectively.

14 `Isave_BAS`: *integer* type. If `Isave_BAS = 1` the harmonic oscillator basis wavefunctions are saved in a file named `ho_basis_N`*`dimbasis`*`.dat` with the format

```
x  ho_1(x) ho_2(x) ... ho_dim(x)
```

The unit involved is unit 70.

15 `I_sumr`: *integer* type. If `I_sumr` is nonzero (and positive) the sum rules for the total strength is computed for the first `I_sumr` eigenstates and the energy weighted strength is computed for the ground state. Results are displayed.

16 `I_toten`: *integer* type. If `I_toten` is nonzero (and positive) the B1 and B2 values of the transitions of the first `I_toten` eigenstates to the rest of the states are computed.

17 `B_analytical`: *logical* type. If `B_analytical` is true the B1 and B2 values of the transitions of the first `I_toten` eigenstates to the rest of the states are computed using the HO properties as a matrix product.

18 `B_numerical`: *logical* type. If `B_numerical` is true the B1 and B2 values of the transitions of the first `I_toten` eigenstates to the rest of the states are computed using a numerical integration algorithm.

19 `IPRINT`: *integer* type. Program output verbosity. Mainly for debugging purposes. Possible values in the interval 0-5.

3.1.1 Program Execution

We run the case of a Woods-Saxon potential using the input file given above. Assuming the input file is in the `test` directory we run the file with the command

```
test$ ../bin/wsaxon_1body_1D_HO < harm_osc_1D_1body_wsaxon.inp
```

An excerpt of the output is as follows

```
 X grid step =   0.18641810918774968      fm
HARMONIC FUNCTION            1  NORMALIZATION   1.0000000000000000
HARMONIC FUNCTION            2  NORMALIZATION   1.0000000000000002
.
.
.
HARMONIC FUNCTION          500  NORMALIZATION  0.99999999999999922
HARMONIC FUNCTION          501  NORMALIZATION   1.0000000000000013
         44  EIGENVALUES COMPUTED IN A          500  DIM HARMONIC BASIS
          1  -39.575743429858143
          2  -17.870764051008919
          3 -0.51376864785484155
          4   2.2161186319837708E-002
          5   2.5262705152647863E-002
          6   8.8640984430509134E-002
          .
          .
          .
         42   8.8266692344274524
         43   9.2066002755550400
         44   9.7283452731211177
B1 :: Analytical method, state           1
   1-th state energy:  -0.39575743E+02   <  1| X |Avec(  1)> =    0.78354220E-16
   2-th state energy:  -0.17870764E+02   <  1| X |Avec(  2)> =   -0.98820380E+00
   3-th state energy:  -0.51376865E+00   <  1| X |Avec(  3)> =   -0.15379148E-15
```

```
    .
    .
    .
  42-th state energy:   0.88266692E+01   <  1| X |Avec( 42)> =    0.75484771E-02
  43-th state energy:   0.92066003E+01   <  1| X |Avec( 43)> =    0.20615985E-14
  44-th state energy:   0.97283453E+01   <  1| X |Avec( 44)> =    0.71621324E-02
Total B1:            1  0.97756688413360637
B1 :: Analytical method, state            2
   1-th state energy:  -0.39575743E+02   <  2| X |Avec(  1)> =   -0.98820380E+00
   2-th state energy:  -0.17870764E+02   <  2| X |Avec(  2)> =   -0.23754587E-14
   3-th state energy:  -0.51376865E+00   <  2| X |Avec(  3)> =    0.12096876E+01
    .
    .
    .
  43-th state energy:   0.92066003E+01   <  3| X |Avec( 43)> =   -0.20346665E-13
  44-th state energy:   0.97283453E+01   <  3| X |Avec( 44)> =    0.22883743E+00
Total B1:            3   34.162824527021456
B2 :: Numerical method, state            1
   1-th state energy:  -0.39575743E+02   <  1| X^2 |Avec(  1)> =    0.97792289E+00
   2-th state energy:  -0.17870764E+02   <  1| X^2 |Avec(  2)> =    0.19534173E-14
   3-th state energy:  -0.51376865E+00   <  1| X^2 |Avec(  3)> =   -0.10810122E+01
    .
    .
    .
  43-th state energy:   0.92066003E+01   <  1| X^2 |Avec( 43)> =    0.14978555E+00
  44-th state energy:   0.97283453E+01   <  1| X^2 |Avec( 44)> =   -0.88318993E-14
Total B2:            1   2.7847318688712841
B2 :: Numerical method, state            2
   1-th state energy:  -0.39575743E+02   <  2| X^2 |Avec(  1)> =    0.19534173E-14
   2-th state energy:  -0.17870764E+02   <  2| X^2 |Avec(  2)> =    0.31915150E+01
   3-th state energy:  -0.51376865E+00   <  2| X^2 |Avec(  3)> =    0.50963500E-13
    .
    .
    .
  43-th state energy:   0.92066003E+01   <  2| X^2 |Avec( 43)> =    0.63722266E-11
  44-th state energy:   0.97283453E+01   <  2| X^2 |Avec( 44)> =    0.60894549E+00
Total B2:            2   16.285900094324546
B2 :: Numerical method, state            3
   1-th state energy:  -0.39575743E+02   <  3| X^2 |Avec(  1)> =   -0.10810122E+01
   2-th state energy:  -0.17870764E+02   <  3| X^2 |Avec(  2)> =    0.50963500E-13
   3-th state energy:  -0.51376865E+00   <  3| X^2 |Avec(  3)> =    0.34446553E+02
    .
    .
    .
  43-th state energy:   0.92066003E+01   <  3| X^2 |Avec( 43)> =    0.86716506E+00
  44-th state energy:   0.97283453E+01   <  3| X^2 |Avec( 44)> =   -0.43781858E-13
Total B2:            3   4395.9771563158238
```

The output is quite verbose for the given options. Note that the onset of the output for `IPRINT` larger than one are the display of the main parameter values and a check of the HO basis normalization. This should be checked whenever either the basis dimension or the spatial grid parameter change their values.

The output proceeds with the system energies, with negative values for the bound states and positive values for the discretized pseudostate continuum.

The program then computes the B1 and B2 values for the three bound states as `I_toten = 3`. In each case depending on the values of the logical variables `B_analytical` and `B_numerical` two possible outputs can be displayed; one computed with the numerical integral (called `Numerical method`) and the other algebraically using the HO properties (calle `Analytical method`). Both should coincide.

The following files are saved, note that the `N500` part of the name varies with the `N` value and that in this case there are three bound states and we are asking for `B1` and `B2` results for the three eigenstates.

1 `ho_eigenvalues_N500.dat`: system eigenvalues with values less than `Max_aval_har` in case `Max_aval_har` is not zero.

2 `ho_pot_eigvec_N500.dat`: computed system eigenfunctions plus the energy as a function of `x`.

3 `ho_pot_eigvec2_N500.dat`: squared modulus of the computed eigenfunctions plus the energy as a function of `x`.

4 `ho_E1_N500_i.dat, i = 1, 2, 3`: Two columns, `E_k |<i|x|k;>|^2` for `k = 1, ..., dim_C`

5 `ho_E1_TM_N500.dat`: Three columns, `E_k <1|x|k> <2|x|k>| <3|x|k>` for `k = 1, ..., dim_C`

6 `ho_E2_N500_i.dat, i = 1, 2, 3`: Two columns, `E_k |<i|x^2|k>|^2` for `k = 1, ..., dim_C`

7 `ho_E2_TM_N500.dat`: Three columns, `E_k <1|x^2|k> <2|x^2|k>| <3|x^2|k>` for `k = 1, ..., dim_C`

The value `dim_C` is equal to the truncated basis dimension `Dim_HO` if `Max_aval_har` is zero and to the number of eigenstates with eigenvalues less than `Max_aval_har` otherwise.

Chapter 4

References

1 Minuit minimization package (http://www.cern.ch/minuit): Comput. Phys. Commun. *10* (1975) 343-367.

2 J. Chem. Phys. *104* (1996) 6956.

3 Chem. Phys. Lett. *365* (2002) 57-68.

4 Chem. Phys. Lett. *375* (2003) 309-320.

5 Phys. Rev. *A77* (2008) 032115.