EJ_Calc

A code for calculation of exchange coupling constants using spin densities

Contents

1   Introduction

Briefly, this code calculates coupling constants using the following Hamiltonian:[1-3]

$$\hat{H} = -2\sum_{i>j} J_{ij} s_i s_j \tag{1}$$

Where J_{ij} is the exchange coupling constant, and s_i and s_j are the spin operators. Any definition of spin operators (formal spins or spin projections or spin densities or any other) can be used. Sometimes researchers exclude the pre-factor 2 in the above formula so this needs to be kept in mind while using this code.

2   Theory

For obtaining n coupling constants, at least (n+1) spin configurations need to be modelled. In principle, for x centres with unpaired electrons, there can be a maximum of x(x-1)/2 coupling constants and a total of 2^x different spin configurations. Out of these 2^x configurations, half would be the mirror image of the others. For example, for any Mn_6^{III} system one can have a maximum of (6 X (6-1) / 2 =) 15 coupling constants while the total number of spin configurations for this complex are (2^6 =) 64 out of which a maximum of only 32 can be unique.

For any system with more than 3 paramagnetic centres, the number of configurations that are available are more than that required to calculate all coupling constants. This can be used to our advantage since we can model more states than required. This allows us to remove any dependence of the calculated coupling constants on the states modelled.[4-8] However, depending upon the size of the system and the number of states modelled, one can end up with a large number of sets of solutions (coupling constants) which have to be averaged in some sensible way. This problem has been addressed in this code.

When calculating *r* coupling constants using *n* (where *n > r*) spin configurations, then first of all, one of these configurations has to be used as a reference for all the other states so that we can determine the energy only related to the flipping of spin. With the remaining $(n-1)$ configurations, a total of $\left[\frac{(n-1)!}{r!\,(n-1-r)!}\right]$ (i.e. $^{(n-1)}C_r$) sets of solutions (i.e. r coupling constants) are obtained. Since any of the n modelled spin states can be used as a reference, $n\left[\frac{(n-1)!}{r!\,(n-1-r)!}\right]$ (i.e. n($^{(n-1)}C_r$)) sets of solutions are obtained.

This code takes the Hamiltonian, the values of spin operators for each paramagnetic centre in each modelled spin state and the energy of each modelled spin state as input. This information is then used to calculate the coupling constants. As mentioned earlier, when calculating *r* coupling constants using *n* (where *n > r*) spin configurations, a total of n($^{(n-1)}C_r$) sets of coupling constants are obtained. The sets which turn out to be singular are discarded. The valid solutions are averaged and the standard deviation is calculated. The solutions that are deviating by more than three standard deviations are discarded and the standard deviation is calculated again and this cycle is repeated until self-consistency is obtained.

The coupling constants obtained this way are then used to further determine the energy of all the possible electronic states ($2^{0.5x}$ for x paramagnetic centres). To achieve this, the Heisenberg Dirac van

Vleck Hamiltonian (eq. 1) is used. The spin operator value for each paramagnetic centre is determined by taking an average of the spin operator value in the different electronic states that is provided as input. Two electronic states are considered to be degenerate if the difference in the energy between the two is less than 1 cm^{-1}.

3   Basic details

This code has been written in FORTRAN 2008 and has two executables – ej_calc_form and ej_calc. The calculation of coupling constants using spin densities requires the use of both executables and the execution of ej_calc_form should be followed by execution of ej_calc. Each executable requires two input files – one specifying the Hamiltonian for the system under study and another that contains the definition of spin operator (i.e. the spin density) for each paramagnetic centre and the energy of the system in the given spin state.

The executable ej_calc_form makes use of formal spin values to determine the j-values while ej_calc uses the actual value of spin operators for the same purpose. This has been done because the value of spin operators are generally not whole numbers and it is difficult to identify singular sets of equations without the use of whole numbers. Since the actual values of spin operators are generally not far off from the nearest whole number, it is highly unlikely that the use of spin operators will remove the singularity. ej_calc_form produces 4 output files – a .txt file containing the coupling constants and 3 other files – bigg, nrg_frm_jval_form and singul. ej_calc produces 2 output files - a .txt file containing the coupling constants and nrg_frm_jval. The details about the coupling constants is printed in the .txt file. Information contained in the other files is described in the following discussion.

One starts by executing ej_calc_form which rounds the spin operator values to the nearest whole number and uses the corresponding whole number as the value of the spin operator. It identifies which sets of equations are singular, discards them and stores information about these sets in the file 'singul'. In rare occasions, equations are encountered which are close to singularity thereby leading to absurd (non-infinite) solutions. To keep note of these equations, details are printed in the file 'bigg'. The criteria to identify such equations has been set such that any coupling stronger than 50 cm^{-1} is considered too big. These solutions are not discarded as in some cases coupling constants can be quite large. If, however, they are outliers, they will be removed eventually since at the end, only those solutions will be considered valid which lie within 3 standard deviations of the average value as mentioned previously. ej_calc reads the 'singul' file generated by ej_calc_form and calculates the solutions for only the sets that were found to be non-singular by ej_calc_form.

The files nrg_frm_jval_form and nrg_frm_jval contain the approximate energy of the all possible $2^{0.5x}$ states (x = number of paramagnetic centres). The energy is calculated using eq. 1. The spin operators for all the states are defined by taking the average of the value of spin operator for each paramagnetic centre that was provided as input. Since the variation of the value of spin operators in different spin states is small, the energy obtained this way serves as a reasonable approximation to the actual energy of these states. The true energies of all spin states can, of course, be obtained by diagonalising the full Hamiltonian. Since this diagonalisation becomes computationally quite expensive for large systems,

the method employed in this code provides an inexpensive way to obtain a good approximation of the energy of these states.

4 For Users

The execution of the code needs to be done in two steps.

(i)   The first step is to execute ej_calc_form

./ej_calc_form "file 1" "file 2"

As mentioned before, it is necessary to run ej_calc_form to produce the "singul" file which is required to calculate the coupling constant using ej_calc.

(ii)   This is followed by the execution of ej_calc as:

./ej_calc "file 1" "file 2"

"file 1" is the file containing the definition of the Hamiltonian

"file 2" is the file containing the value of spin operators for each paramagnetic centre and the energy of different spin states

The details about the coupling constants are printed in the files "file 1"_form.txt and "file 1".txt for ej_calc_form and ej_calc respectively.

The details of the format for file 1 and file 2 and an example of each are provided in the following discussion.

magnetic centres

<number of paramagnetic centres>

J values

<number of j-value>

Hamiltonian

<First interaction associated with the first j-value>

<Second interaction associated with the first j-value>

<Third interaction associated with the first j-value>

And so on

****

<First interaction associated with the second j-value>

<Second interaction associated with the second j-value>

<Third interaction associated with the first j-value>

And so on

****

<First interaction associated with the third j-value>

<Second interaction associated with the third j-value>

<Third interaction associated with the first j-value>

And so on

Hamiltonian Ends

This is the format of the input file.

The first line has to state 'magnetic centres' and the second line will specify how many magnetic centres are there. Then the number of J-values are specified. After that the Hamiltonian is specified.

The spin interactions under a single J-value are specified in a single block. The spin interactions for different J-values are separated by '****'. The line 'Hamiltonian Ends' is needed in the end and is not preceded by '****'.

Note: The terms 'magnetic centres', 'J-values', 'Hamiltonian' and 'Hamiltonian Ends' are case-sensitive.

<S1> <S2> <S3> ………. Energy of the system in spin state 1 (Hartrees)

<S1'> <S2'> <S3'> ………. Energy of the system in spin state 2 (Hartrees)

<S1''> <S2''> <S3''> ………. Energy of the system in spin state 2 (Hartrees)

…..

….

Here <Si>, <Si'> and <Si''> denote the spin operator value for the ith paramagnetic centre in the given electronic state.

Examples of both input files are provided in the following sections.

4.1 Example Input File for "file 1"

To understand the construction of the input files, let us take the example of a {Mn_6} coordination complex[8] shown in **Figure 1**. The Mn centres in this complex interact with each other via the phosphonate ligands and the central Cl^- ion. Each Mn centre in this complex has 1 trans- and 4 cis-neighbours and one requires 2 J-values to account for these cis- and trans- interactions between Mn centres.[8] Therefore, the Hamiltonian for this complex can written as follows:

$$
\begin{aligned}
H = \ &-2J_1[\langle s_1.s_4\rangle + \langle s_2.s_5\rangle + \langle s_3.s_6\rangle] \\
&-2J_2[\langle s_1.s_2\rangle + \langle s_1.s_3\rangle + \langle s_1.s_5\rangle + \langle s_1.s_6\rangle + \langle s_2.s_3\rangle + \langle s_2.s_4\rangle + \langle s_2.s_6\rangle + \langle s_3.s_4\rangle \\
&+ \langle s_3.s_5\rangle + \langle s_4.s_5\rangle + \langle s_4.s_6\rangle + \langle s_5.s_6\rangle]
\end{aligned}
$$

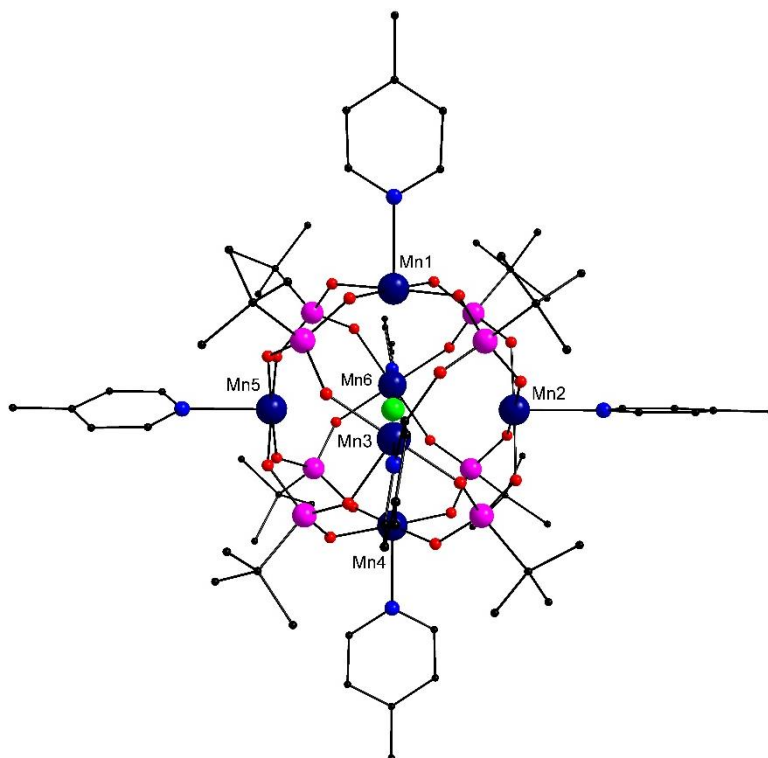**Figure 1:** Structure of the {Mn_6} complex. Colour scheme: Mn (dark blue), P (pink), Cl (green), C (black), N (blue) and O (red). Hydrogen atoms have been removed for clarity.

This Hamiltonian can be represented using the following input file:

```
magnetic centres
6
J values
2
Hamiltonian
1 4
3 6
2 5
****
1 2
1 3
1 5
1 6
2 3
2 4
2 6
3 4
3 5
4 5
4 6
5 6
Hamiltonian Ends
```

4.2 Example Input File for "file 2"

The following is an example spin operator file for the above mentioned Mn_6 system. The spin operators were obtained using Bader population analysis and the energy was obtained from DFT calculations.[8] The first six numbers on each line represent the spin density on each of the six Mn centre in a given spin configuration and the last number on the line represents the energy of the corresponding state in Hartrees.

-3.7863320 3.7861321 3.7863107 3.7868157 -3.7862305 -3.7862343 -8.605807229691483E+03
-3.7861476 3.7886816 3.7866279 3.7867024 3.7888073 -3.7859959 -8.605807007350464E+03
-3.7885975 3.7885472 3.7866686 -3.7885253 3.7885196 -3.7865706 -8.605806946257479E+03
-3.7883188 3.7884830 3.7887427 -3.7882102 3.7889254 3.7890060 -8.605806714840277E+03
-3.7861387 3.7889810 3.7892562 3.7871543 3.7892340 3.7891449 -8.605806609885272E+03
3.7890715 3.7894288 3.7894591 3.7895166 3.7895544 3.7891216 -8.605806049221312E+03

The energies are given here in scientific notation but they can be in non-scientific notation too.

These example input files and the resultant output file have also been provided as separate files in the package.

5   Description of the output file (file1.txt)

The format of the output file is as follows:

Output file sections	Description
Number of Magnetic centres: 6 Total number of possible interactions: 15 Input obtained from spin density file ……… ……… Number of spin density sets: 6 Spin Matrix ……… ………	Details of the basic input parameters and the information obtained from the file containing the spin density information are given here.
No. of J value asked for = 2 Interactions considered under each J value J 1 1 2 1 3 1 5 1 6 2 3 2 4 2 6 3 4 3 5 4 5 4 6 5 6 J 2 1 4 3 6 2 5 Coefficients of J values ……… ……… ………	The interactions accounted for by each J-value are provided here. The coefficients associated with each J-value based on the Hamiltonian are also given.

Possible combination of equations per set of equations
 considered (i.e. total number of equations -1):
        5 C   2 =   10
Solving the equations
 Average J values

Ref. eq.	non-singular solutions	J 1	J 2
1	10	-3.4459	-1.2797
2	10	-3.4757	-1.2704
………			
………			
………			

The total possible combinations after using one of the equations as a reference is mentioned here. This is followed by the details of the average J-values obtained for each set with each set differing in the equation used as a reference.

Removing solutions in which J-values deviate by more than 3 standard deviations
 Standard deviation (percentage standard deviation) on different J-values:
non-singular equations   J 1   J 2
………..

Information on standard deviations for each J-values when cycling through all valid solutions to ensure that each of them is within 3 standard deviations is given here.

Final results
 Total non-singular equations   60
 Average J-values (cm-1) and standard deviations
          J 1   J 2
J-val   -3.4803   -1.2787
Std. Dev. 0.05(1%) 0.04(3%)

Summary of results.

Comparison of energy (cm-1) of electronic states calculated by DFT
 and energy obtained using the average J-values calculated:
(Note: The following are the energies assuming the first
 energy value in the spin density file as the reference value)
 DFT Energy(cm-1)   Calculated energy(cm-1) Absolute Difference(cm-1)
    0.00000   0.00000   0.00000(NaN%)
………
………
………

A comparison between the energy of the different states calculated using DFT and that calculated using the J-values is provided in this section. This can be used to check how well the coupling constants describe the different states. Large differences between the two energy values indicate problems in either the modelled states or the Hamiltonian used.

6   Summary of files

A summary of all input and output files is given below:

Filename	File type	Function
file1	Input	Contains information about total number of paramagnetic centres and the Hamiltonian specifying the relationship between various metal centres.
file2	Input	Contains spin densities on the paramagnetic centres in different spin states and the energy of the corresponding states.
file1.txt	Output	Contains information about the calculated coupling constants.
bigg	Output	Contains information about sets of equations that yield J-values > 50 cm^{-1}.
singul	Output	Contains information about sets of equations that are singular and is produced by ej_calc_form. This file is required for the proper execution of the executable ej_calc.
nrg_frm_jval	Output	Contains the approximate energy of the all possible $2^{0.5x}$ states (x = number of paramagnetic centres).

7 For Developers

7.1 Workflow

Both ej_calc_form before ej_calc are quite similar in their workings. The code, in both cases is divided into 4 modules:

1. main.f90: controls the flow of the code.
2. init.f90: declares all the global variables, reads the input files and initializes the global variable using the information obtained from the input files.
3. modul_lib.f90: builds up the Hamiltonian based on the definition obtained from the input file. It further forms different combinations of the equations and solves them using LAPACK routines. The average of all the valid sets of j-values and the standard deviations for each J-value is also determined using this module.
4. enrg_frm_jval.f90: calculates the energy of the all spin states using eq.1.

7.2 ej_calc_form

A detailed breakdown of ej_calc_form is provided below.

7.2.1 Init.f90

This module defines all the global variables.

Variable	Function
num_mag_cent	stores the number of metal centres in each configuration.
no_of_j_val	stores the total number of j-values defined for the system.
tot_interac	stores the number of possible interactions between spins i.e. the total number of S1S2 terms.
num_spin_dens_set	stores the number of sets of spin density available which is the total number of lines in the spin density file.
try, try1	these in combination help in keeping the program running just in case there is an equation set is encountered that cannot be solved.
poss_comb	stores the possible combination of equations i.e. the $^{(n-1)}C_j$ term.
main_cntr	keep track of total non-singular equations.
start, finish	These in combination determine the time taken during the program execution.

Array	Function
hamil1, hamil2	These in combination keep track of which centres are interacting .
jpos	keeps track of how many interactions are being included under a single j-value.
energy	stores the energy for each electronic configuration.
tot_avg	stores the average of all the J-value sets which are valid and which meet the standard deviation criteria.
stddev	stores the standard deviation for the different J-values.
std_per	stores the percentage standard deviation for the different J-values.
spin	stores the spin operator value associated with each metal centre as given in the spin operator file.
hamil	stores the 2S1S2 type weight terms for each interaction.
jval	stores the total (2S1S2 type) coefficient associated with each j-value.
jval_trkr	stores all the computed sets of j-values.

7.2.2   modul_lib.f90

This module contains all the subroutines required to calculate all the possible sets of j-values and determine the valid ones based on standard deviation. The following subroutines form part of this module:

1. **jvals**: This subroutine will determine the S1S2 terms and which of these terms comes under a given j-value.
2. **solv**: This subroutine forms all possible sets of equations and solves them. The local variables used in this subroutine are as follows:

Array	Function
ctr	allows the separation and hence the evaluation of all possible combinations.
loc_energy	stores the energy locally since that will be changed in each set of equations considered.
enrg_tb_slvd1	local copy of the global array 'energy'.
enrg_tb_slvd	stores the energy of the set of equations that will be solved in one instance.
enrg_tb_slvd2	copy of the array 'enrg_tb_slvd'.
loc_jval	store the coefficients of j-values locally since that will be changed in each set of equations considered.
jval_tb_solvd1	local copy of the global array 'jval'.

jval_tb_solvd	stores the coefficients of j-values of the set of equations that will be solved in one instance.
jval_tb_solvd2	copy of the array 'jval_tb_solvd'.

In this subroutine, first of all, an electronic state is chosen as the reference state (every state is chosen as a reference state once) and a new set of equations is formed by subtracting the reference state from all the other states. These states are then grouped into all possible sets of r (the number of j-values one is looking for) and each such set is solved as simultaneous linear equations using the LAPACK library routine dgesv. The LAPACK routine dgesv is implemented in a different subroutine called simul_solv which also performs a few basic checks for singularity. If the solution is found to be singular, the value of all j-values for the given set is set to zero. If, however, the set is not found to be singular, a more thorough screening of the set is performed using the LAPACK library routine dgesvx which is implemented in the subroutine simul_solv1. Additionally, if any of the j-value is found to be larger than 50 cm^{-1} for a given set of equations, details about the set and the corresponding J-values obtained are printed in the file bigg.

3. **checkpos**: This subroutine is a helper subroutine for the subroutine 'solv' to cycle through all the possible combinations of r (the number of j-values one is looking for) spin states from n (or more accurately n-1) spin states.
4. **simul_solv**: This subroutine is also a helper subroutine for the subroutine 'solv'. It checks if there are any identical pair of equations which will automatically make the whole set singular. If such pairs are not found, then a solution is calculated using the LAPACK library routine dgesv.
5. **simul_solv1**: This subroutine is also a helper subroutine for the subroutine 'solv'. It calculates the solution for the same set of equations that were considered solvable by simul_solv using the LAPACK library routine dgesvx which looks further for sets of equations close to singularity. The reason why I use this only for the sets which are found to be solvable by dgesv because it is more computationally expensive (The speed gain in keeping simul_solv and simul_solv1 has not been determined and considering that this code can still solve ~1,000,000 sets of equations within minutes, it was not considered worth doing).
6. **soln_chk**: This subroutine makes sure that the code does not get stuck on any set of equations.
7. **avrg_calcs**: This subroutine calculates the average of all the valid set of solutions.
8. **std_dev**: This subroutine calculates the standard deviation for all the valid set of solutions.
9. **std_dev1**: This subroutine checks if all the valid j-value sets are within 3 standard deviations or not and if they are not, they are removed.
10. **avrg_calcs1**: This subroutine will calculate the new average for j-values since some have been eliminated by the subroutine 'std_dev1'.
11. **final_print**: This subroutine prints the final set of J-values.
12. **backtrack1**: This subroutine will calculate the energies using the computed J-values and compares them with the energies that are provided as input.

7.2.3 enrg_frm_jval.f90

This module calculates the energy of the spin states using the coupling constants calculated by modul_lib.f90. The main local variables used in this module are as follows:

Array	Function
unp_elec	stores the spin operator value for each metal centre.
all_config	stores all the possible spin configurations of spin states.
new_nrg	stores the energy of all possible spin states.

Variable	Function
poss_comb1	stores the total possible spin states.

The following subroutines form part of this module:

1. **avg_spin**: This subroutine calculates the average spin operator (in this case, the formal spin) value for each metal centre.
2. **all_poss_spins**: This subroutine along with the subroutine 'update_spin' forms all possible spin configurations, stores them in the array 'all_config', determines the energy of each configuration in accordance with the specified hamiltonian and stores the energy in the array 'new_nrg'.
3. **sort**: This subroutine sorts all the spin states on the basis of increasing energy using bubble sort.
4. **swap:** This is a helper subroutine for the subroutine 'sort'.
5. **prnt:** This subroutine prints the sorted energies in the nrg_frm_jval file.
6. **update_spin**: This is a helper subroutine for the subroutine 'all_poss_spins' for cycling through all possible spin configurations.

7.3   ej_calc

As mentioned previously, this is similar to ej_calc_form. The differences are pointed out below

7.3.1   init.f90

This subroutine declares two additional arrays:

Array	Function
tot_avg	stores the average value of each coupling constant and is used by the module 'energy_frm_jval.f90'.
singul	stores the location of singular sets of equations identified by ej_calc_form. It essentially stores the data in the file singul.

7.3.2   modul_lib.f90

The subroutines simul_solv1 and soln_chk are not used here. The subroutine solv is different from that in ej_calc_form. It forms all possible sets of equations and solves them using the subroutine simul_solv (which uses the LAPACK subroutine dgesv). There are no checks for singularity. Only those sets that are determined as non-singular by the array singul are solved.

7.3.3   enrg_frm_jval.f90

This module contains an additional subroutine std_dev2d which calculates the standard deviation for the spin operator values that are used to calculate the energy of all spin states.

8   References

1.   P. A. M. Dirac, *Proc. R. Soc. Lond. A*, 1929, **123**, 714-733.
2.   W. Heisenberg, *Z. Physik*, 1928, **49**, 619-636.
3.   J. H. Vanvleck, *Rev. Mod. Phys.*, 1945, **17**, 27-47.
4.   E. Cremades, J. Cano, E. Ruiz, G. Rajaraman, C. J. Milios and E. K. Brechin, *Inorg. Chem.*, 2009, **48**, 8012-8019.
5.   E. Cremades, T. Cauchy, J. Cano and E. Ruiz, *Dalton Trans.*, 2009, 5873-5878.
6.   T. Rajeshkumar, H. V. Annadata, M. Evangelisti, S. K. Langley, N. F. Chilton, K. S. Murray and G. Rajaraman, *Inorg. Chem.*, 2015, **54**, 1661-1670.
7.   K. R. Vignesh, S. K. Langley, K. S. Murray and G. Rajaraman, *Chem. Eur. J.*, 2015, **21**, 2881-2892.
8.   S. Tandon, M. Venkatesan, W. Schmitt and G. W. Watson, *Dalton Trans.*, 2020, **49**, 8086-8095.