Edoardo Antonaci 1234431 December 13, 2020

Ising Model

1 Abstract

Our goal is to resolve an Ising quantum model when it describes a many-body quantum system composed by N particles and subject to the action of an external field expressed by λ factor.
The system lives in d=1 dimension and each particle can be in two only different states (spin $\frac{1}{2}$ chain).

2 Theory

A typical approach when studying a many-body quantum system is to use mean field theory. This approach consists to impose that whole quantum system's state is separable in N single particle's state: in other words, we impose that whole system is not entangled. Starting from this point of view we want to consider a quantum Ising model for a system composed by N particles with spin $\frac{1}{2}$ that live in a lattice in d=1 dimension and they are also subject to the action of transverse field whose intensity is described by λ factor.

$$H = \lambda \sum_{i=1}^{N} \sigma_i^z + \sum_{i=1}^{N-1} \sigma_i^x \sigma_{i+1}^x \tag{1}$$

Using mean field approach $|\Psi_{system}\rangle$ becomes :

$$|\Psi_{system}\rangle = |\Psi^{MF}\rangle = \otimes_{i=1}^{N} |\Psi^1\rangle = \otimes_{i=1}^{N} \sum_{\alpha_i=1}^{2} A_{\alpha_i} |\alpha_i\rangle \tag{2}$$

where α_i=(↑, ↓)= 1, 2 and A_{α_i} are coefficients.Using this assumption and looking at thermodynamic limit $(N \to \infty)$,
the *ground state* density energy $e = \frac{E^{MF}_{groundstate}}{N}$ will be function of λ and it will follow this behaviour:

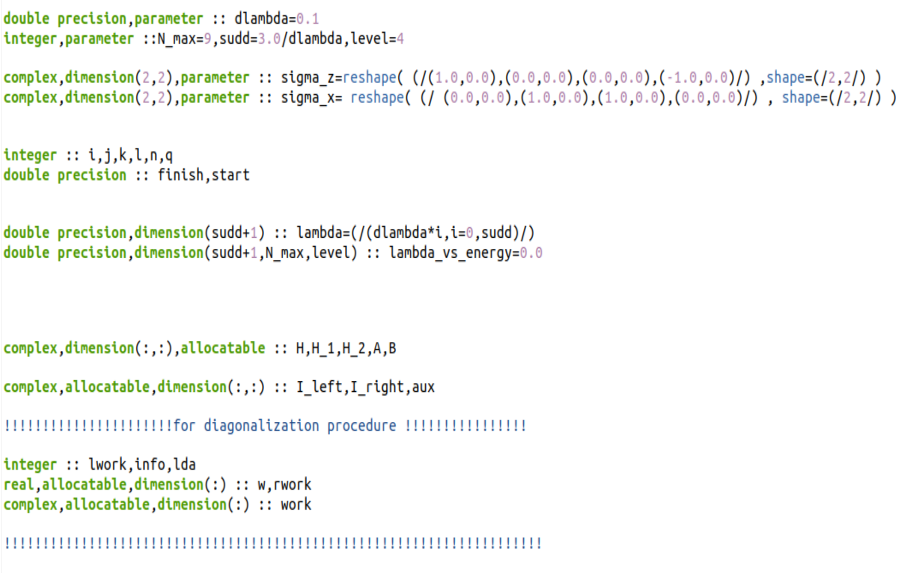
$$e(\lambda) = \begin{cases} -1 - \frac{\lambda^2}{4} & \lambda \in [-2:2] \\ -|\lambda| & \lambda \notin [-2:2] \end{cases} \tag{3}$$

3 Code development

My code is composed by two units program : one *module* and one *main program.*

- PROGRAM ***ex9***:

  here the code does all operations needful in order to "make" an Ising quantum model "by hand" and then we extract from it all infos about energy per particle in different situations.
  At the beginning we impose N_{max} parameter that describes how many particles can be added to model,*level* parameter that describes how many energy states we want investigate and $d\lambda$ parameter that describes how much λ increment will be :
  the code covers interval from λ=0 to λ=3 collecting all values in one array.



```fortran
double precision,parameter :: dlambda=0.1
integer,parameter ::N_max=9,sudd=3.0/dlambda,level=4

complex,dimension(2,2),parameter :: sigma_z=reshape( (/(1.0,0.0),(0.0,0.0),(0.0,0.0),(-1.0,0.0)/) ,shape=(/2,2/) )
complex,dimension(2,2),parameter :: sigma_x= reshape( (/ (0.0,0.0),(1.0,0.0),(1.0,0.0),(0.0,0.0)/) , shape=(/2,2/) )


integer :: i,j,k,l,n,q
double precision :: finish,start


double precision,dimension(sudd+1) :: lambda=(/(dlambda*i,i=0,sudd)/)
double precision,dimension(sudd+1,N_max,level) :: lambda_vs_energy=0.0



complex,dimension(:,:),allocatable :: H,H_1,H_2,A,B

complex,allocatable,dimension(:,:) :: I_left,I_right,aux

!!!!!!!!!!!!!!!!!!!!!!for diagonalization procedure !!!!!!!!!!!!!!!

integer :: lwork,info,lda
real,allocatable,dimension(:) :: w,rwork
complex,allocatable,dimension(:) :: work

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

  Fixed $n \in [2, \dots, N_{max}]$ for each $l \in [0, \dots, 3]$ codes repeats in sequence these following operations :

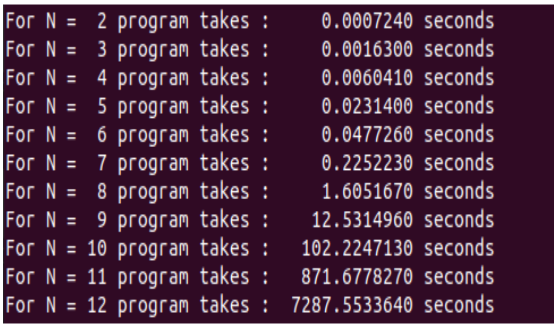
  - executes tensor products with matrices bigger and bigger until $H_1 = \lambda \sum_{i=1}^{N} \sigma_i^z$ and $H_2 = \sum_{i=1}^{N-1} \sigma_i^x \sigma_{i+1}^x$ are correctly defined

  -sums $H = H_1 + H_2$

  -diagonalizes H using *cheev* subroutine (in optimized way) and extract from it w eigenvalues list in ascending order.

  -collects them (normalized by n-1)in a tri-dimensional *lambda-vs-energy* array until last energy level is reached.

  Each time λ-loop is finished ,program prints ,on terminal, time consumed for particular n



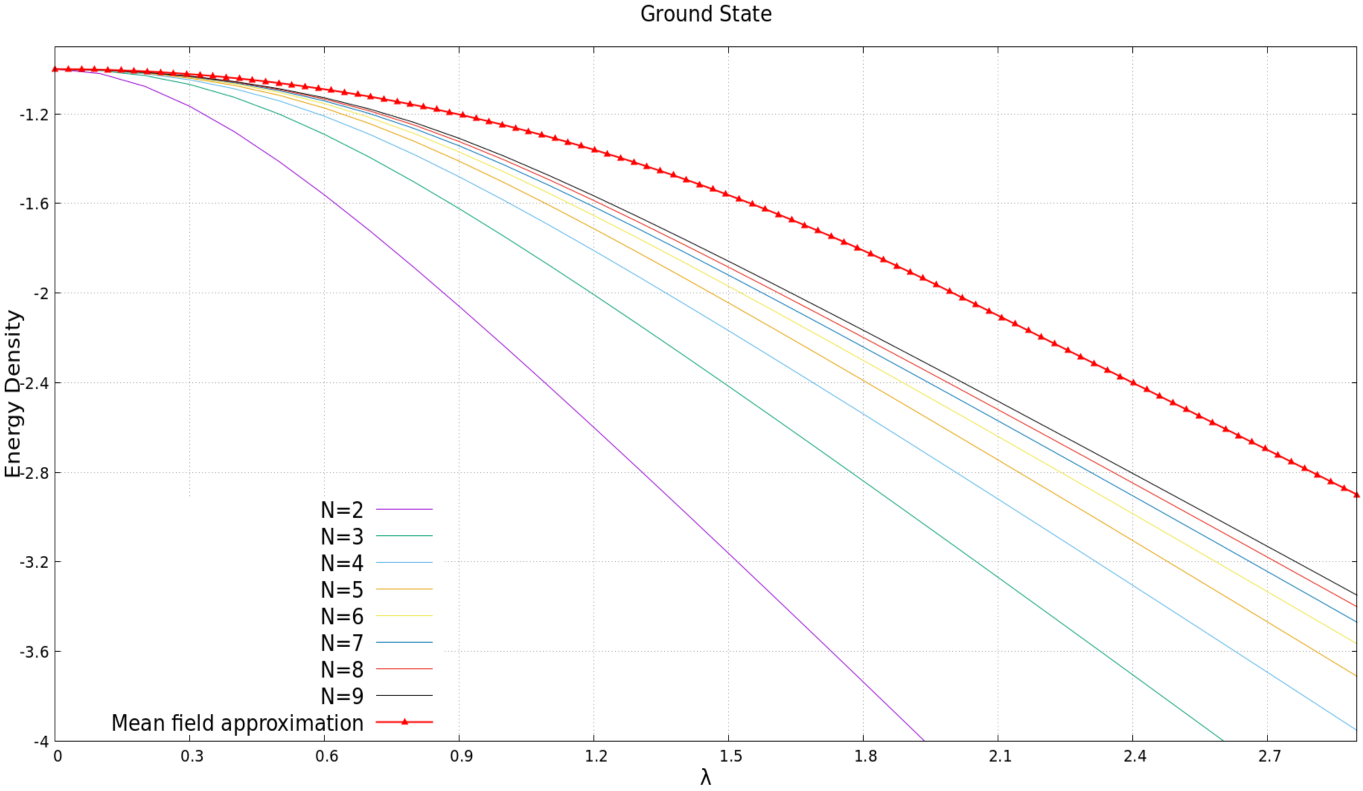
```
For N =  2 program takes :     0.0007240 seconds
For N =  3 program takes :     0.0016300 seconds
For N =  4 program takes :     0.0060410 seconds
For N =  5 program takes :     0.0231400 seconds
For N =  6 program takes :     0.0477260 seconds
For N =  7 program takes :     0.2252230 seconds
For N =  8 program takes :     1.6051670 seconds
For N =  9 program takes :    12.5314960 seconds
For N = 10 program takes :   102.2247130 seconds
For N = 11 program takes :   871.6778270 seconds
For N = 12 program takes :  7287.5533640 seconds
```

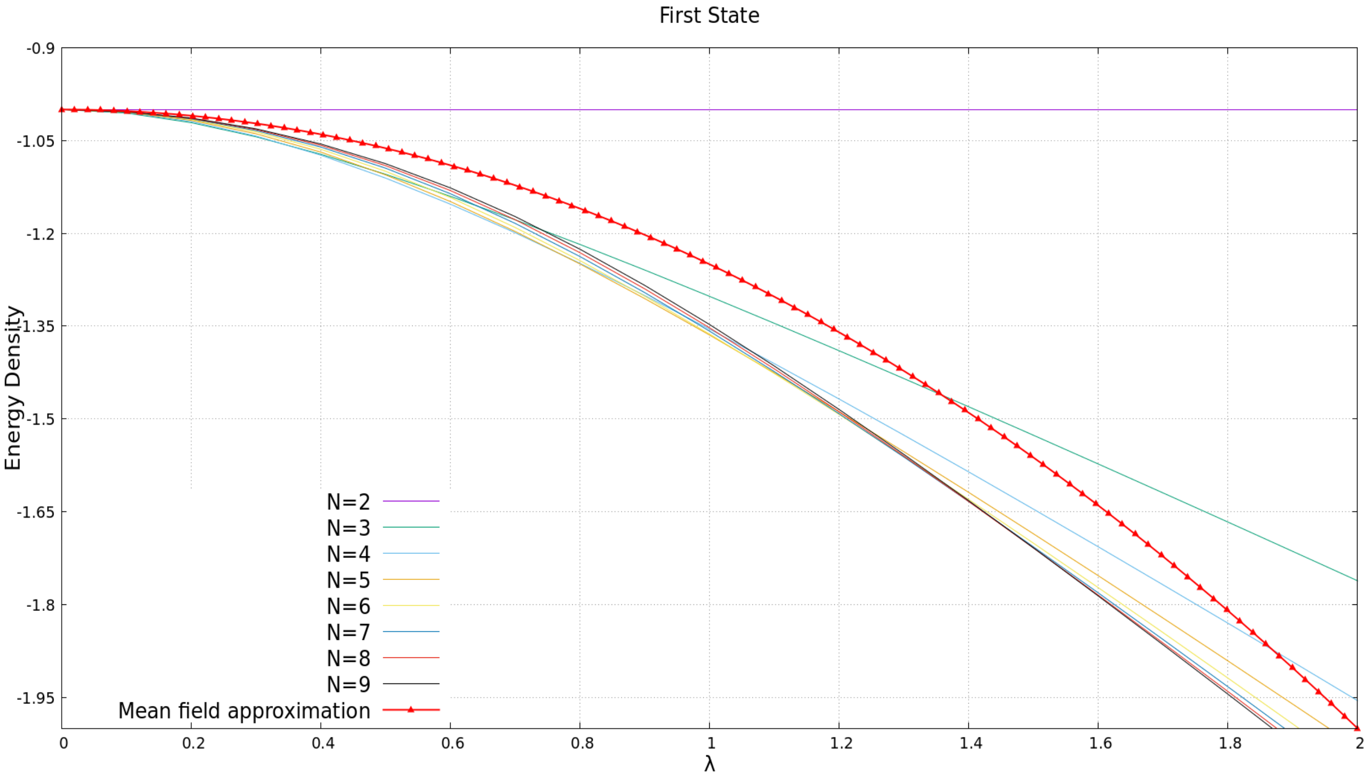
Finally ,for each k-level, code writes on different files the eigenvalues found as function of n (moving by columns) and λ (moving by rows)

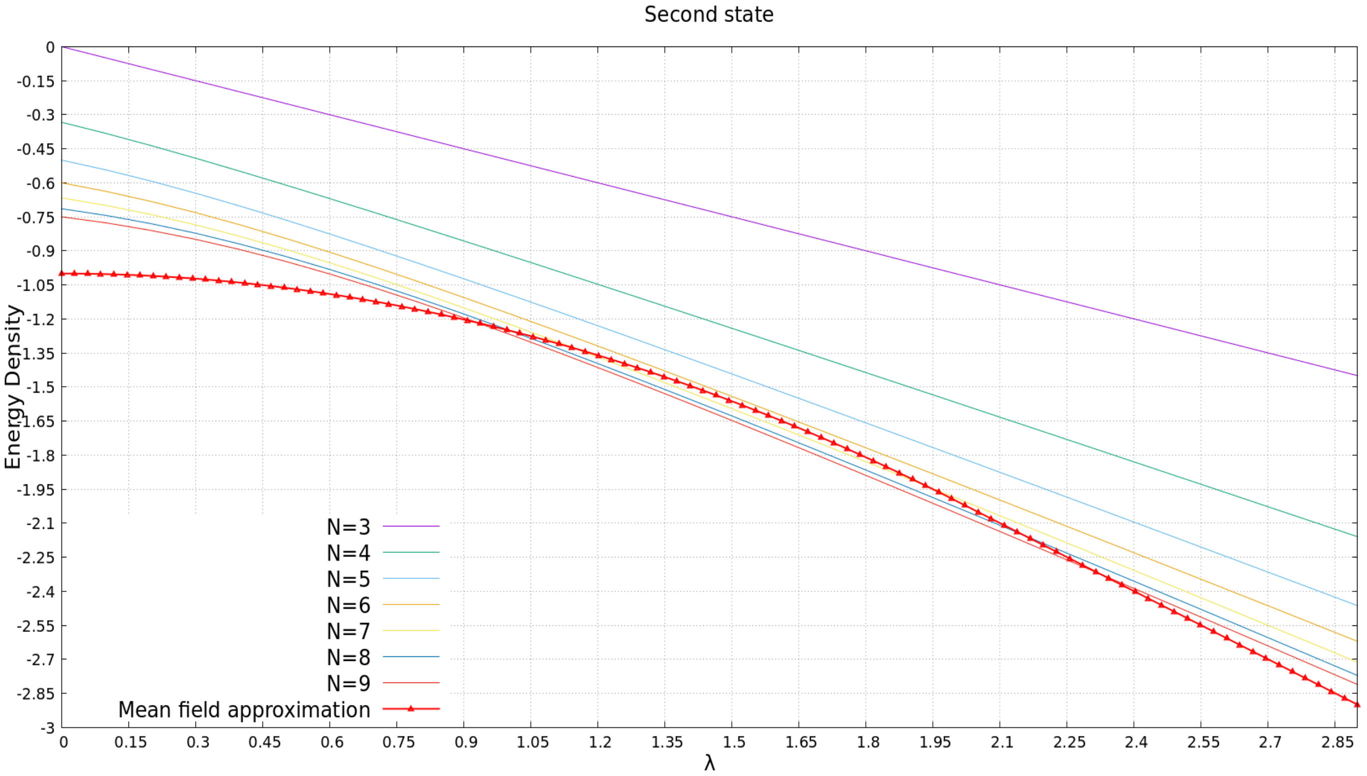
- MODULE ***int-to-string***: here ,in very simple way, code transforms integer number given in input into string and it was very useful in order to optimize writing on files procedure

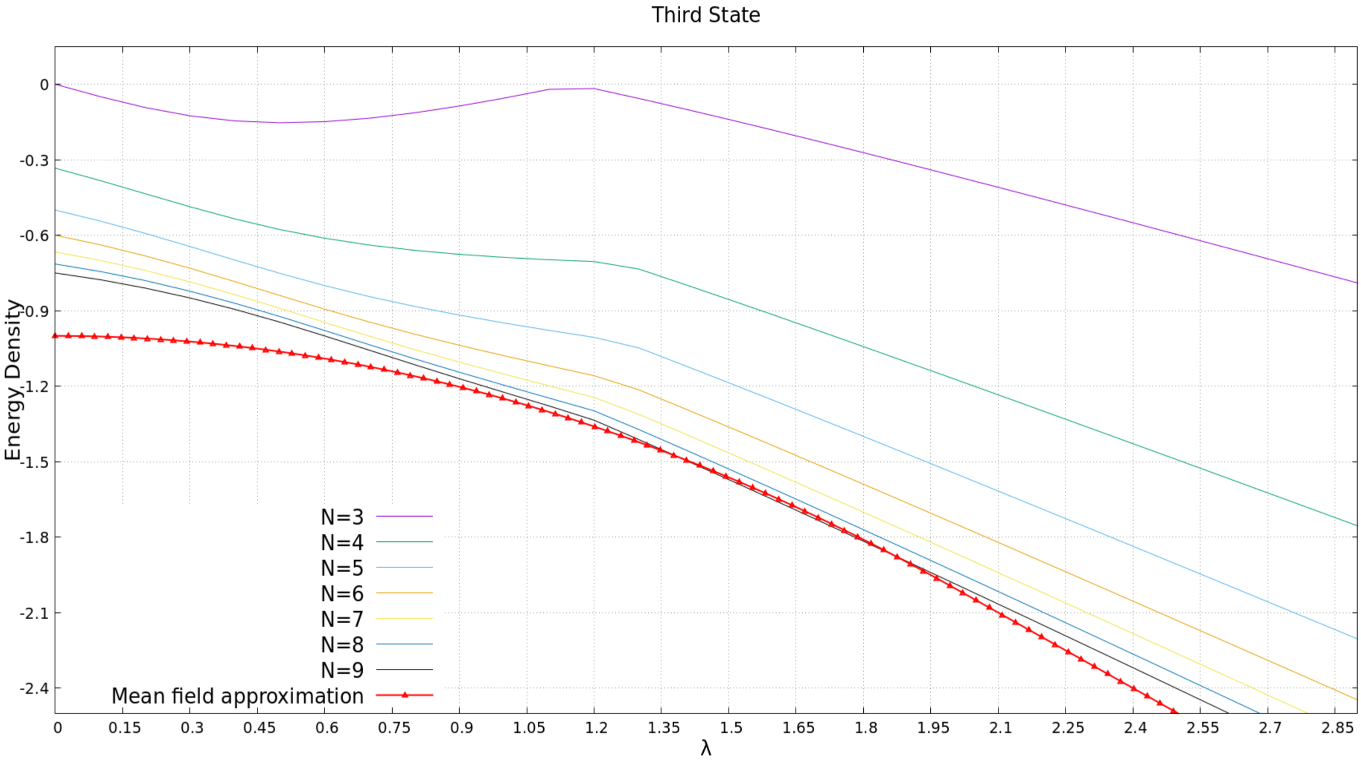
4 Results

In the following we will show results obtained fixing N_{max}=9, $d\lambda = 0.1$ and $level = 4$.




First State
Energy Density
-0.9
-1.05
-1.2
-1.35
-1.5
-1.65
-1.8
-1.95
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2
λ
N=2
N=3
N=4
N=5
N=6
N=7
N=8
N=9
Mean field approximation


Second state
Energy Density
0
-0.15
-0.3
-0.45
-0.6
-0.75
-0.9
-1.05
-1.2
-1.35
-1.5
-1.65
-1.8
-1.95
-2.1
-2.25
-2.4
-2.55
-2.7
-2.85
-3
0
0.15
0.3
0.45
0.6
0.75
0.9
1.05
1.2
1.35
1.5
1.65
1.8
1.95
2.1
2.25
2.4
2.55
2.7
2.85
λ
N=3
N=4
N=5
N=6
N=7
N=8
N=9
Mean field approximation



5 Comments and Self-Evaluation

We decide to plot energy density $\frac{E}{N-1}$ and not $\frac{E}{N}$ because of,being an open chain,first and last particles lose one neighbour. Anyway as we can see ,looking at previous section, two behaviours can be highlighted:

- adding particles to Ising quantum model, ”mean field approximation ” can predict better and better energy density behaviour as function of λ but the prediction is less and less valid augmenting energy level.

  We suppose that it is due to the fact that ”mean field approximation” plotted in graphs ,fits better at thermodynamic limit at only for *ground state*. Moreover when particles are few the main assumption of MF theory ,that is, separability of $|\Psi_{system}\rangle$ has no sense and entanglement effects arise and they are not negligible.

- looking at ”Third State” graph for $\lambda \approx 1.2$ we observe a changing behaviour : after that point energy density behaviour is linear.