Review for MateriApps
Writer: Rico Pohle (2018/10/04)

1. Introduction

In this report I will outline my first experiences of using the linear algebra library **ALPS** (**A**lgorithms and **L**ibraries for **P**hysical **S**imulations), which I made at the the 2nd Innovation Camp for Computational Materials Science (2nd ICCMS) in Yonago, Japan.
Included in this library are quantum solvers, such as ED, DMRG, Quantum Monte Carlo and Matrix Product states. In the assigned group work of this camp we decided to explore the J1-J2 Heisenberg model on the square lattice. My part included calculations of this model with DMRG by using the ALPS library.
Since its installation and compilation is quite time consuming, I used MateriAppsLIVE! on my laptop to access the ALPS library.

2. How to install/compile (if necessary)

I used MateriAppsLIVE! after installation of the Oracle VM virtual box (v5.2.18) on my laptop. The ALPS library is pre-installed in MateriAppsLIVE!, which made its use very simple.

3. How to use

MateriAppsLIVE! starts by loading the virtual box. The environment is a LINUX environment, and allows to access and compute programs in the integrated terminal.
I used the tutorial python scripts, provided at the ALPS web page:
http://alps.comp-phys.org/mediawiki/index.php/ALPS_2_Tutorials:Overview
to get used to the working environment of ALPS.

4. Results

Using ALPS I evaluated the J1-J2 Heisenberg model on a 4x4=16 site square lattice with DMRG. I modified the input files such that:

```
#prepare the input parameters
parms = [ {
      'LATTICE'                    : "frustrated square lattice",
      'MODEL'                      : "spin",
      'CONSERVED_QUANTUMNUMBERS'   : 'N,Sz',
      'Sz_total                    : 0,
      'J0'                         : 1,
      'J1'                         : j1,
      'SWEEPS'                     : 4,
      'NUMBER_EIGENVALUES'         : 1,
      'L'                          : 4,
```

```
    'W'                                          : 4,
    'MAXSTATES'                                  : alpha,
    'MEASURE_LOCAL[Local magnetization]'     : 'Sz'
} ]
```

---

where I modified in each calculation the ratio of next-nearest neighbor to nearest-neighbor exchange interaction J1/J0, and the maximum number α of DMRG states kept in each calculation.

Fig.1 shows the DMRG results of the T=0 ground state energy, while Fig.2 shows the relative error to the exact diagonalization (ED) result obtained, by another member of our group. The large the number of maximal states α kept in each DMRG calculation the better the result compares to the ED result.

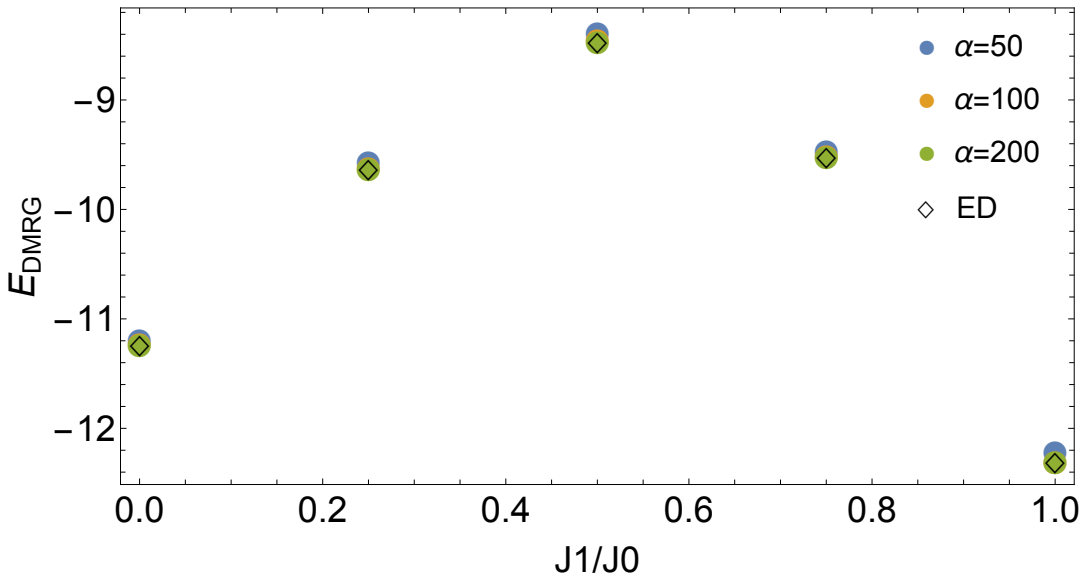

Fig. 1: Ground state energy for the Heisenberg model on the square lattice, obtained by DMRG, as a function of next-nearest neighbor interaction J1/J0 and the maximum number of states kept in each calculation α.

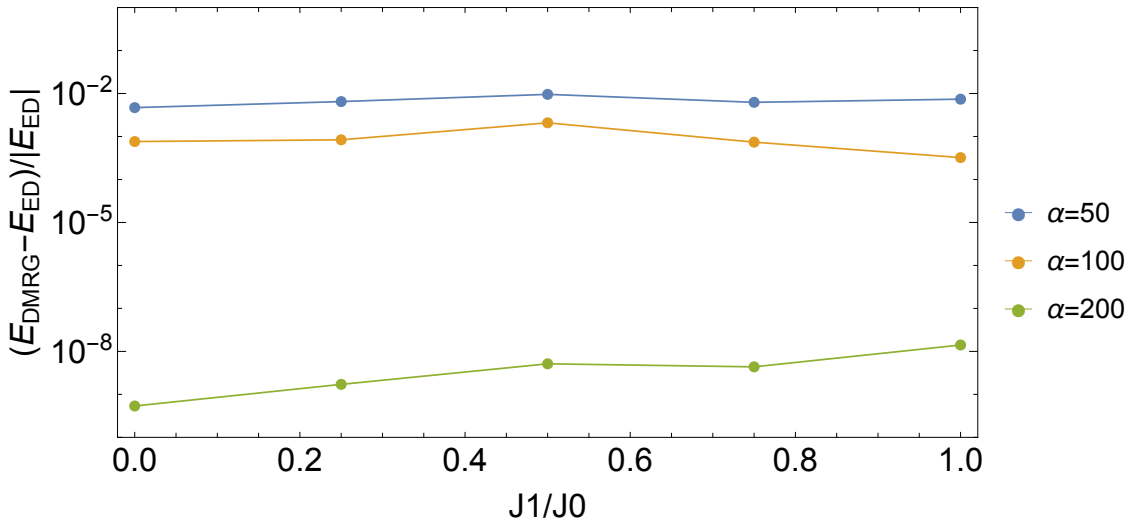

Fig.2: Energy difference of the ground-state energy of the Heisenberg model, between DMRG and ED for different next-nearest neighbor interaction J1/J0 and the maximum number of states kept in each calculation α.

4. Summary

In the 2nd Innovation Camp for Computational Materials Science (2nd ICCMS) I got the opportunity to deepen my knowledge of the linear algebra library **ALPS** (**A**lgorithms and **L**ibraries for **P**hysical **S**imulations).
I used DMRG from this library in the environment of MateriAppsLIVE! as a virtual box to explore the J1-J2 Heisenberg model on the square lattice. We found that the DMRG results match the results of ED if the number of states kept in each DMRG calculation is sufficiently large.