PSL-6

TeNeS

https://pasums.issp.u-tokyo.ac.jp/tenes

TeNeS -- Tensor network solver package for the two-dimensional quantum lattice model

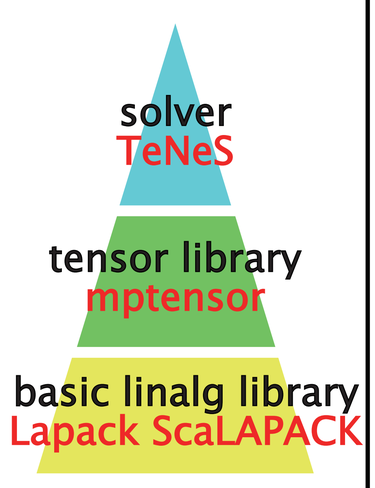
Yuichi Motoyama (ISSP), Tsuyoshi Okubo (UTokyo), Satoshi Morita (ISSP), Naoki Kawashima (ISSP), Kazuyoshi Yoshimi (ISSP), and Takeo Kato (ISSP)

What's TeNeS?

TeNeS (Tensor Network Solver) a two dimensional quantum lattice model solver based on iTPS.
TeNeS offers several widely-used models and lattices, and thus users can perform their simulation easily by preparing **a short simple input file.**
Additionally, this program makes use of a **OpenMP/MPI hybrid parallelized tensor operations library, mptensor**, and thus works on a massively parallel machine well.

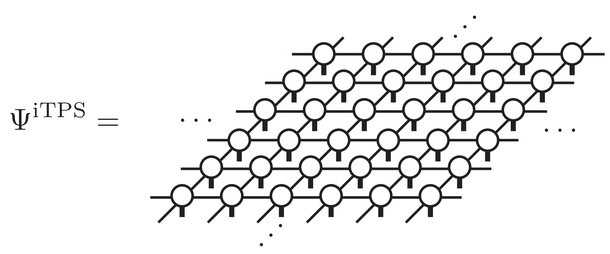
Tensor network method is a promising method and easy to understand conceptually, but is difficult to implement in practice.
We hope that TeNeS becomes the first step of new-comers, and for professionals gives a baseline for developing a new method.

TeNeS is a free/open source software, and available under the GNU GPL v3.
You're welcome to contribute/fork!
https://github.com/issp-center-dev/TeNeS

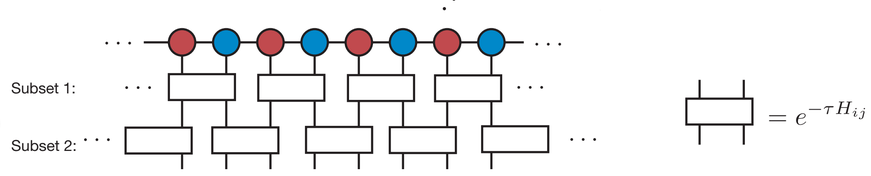


Methods in TeNeS

iTPS on a square lattice as a wavefunction

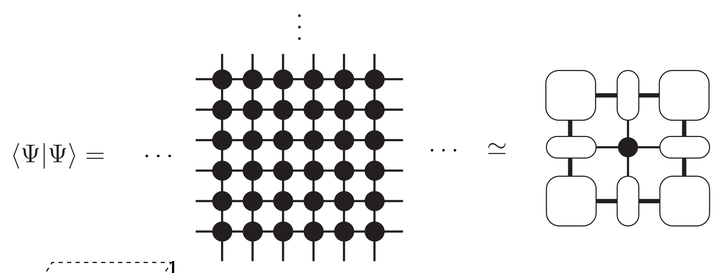


Optimization of iTPS by imaginary time evolution (ITE)

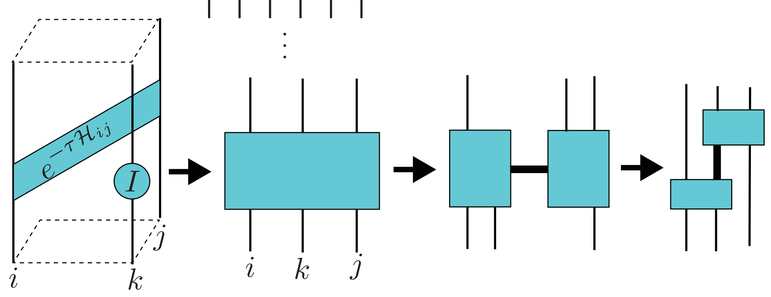


[**simple update (SU)** and **full update (FU)** are implemented]

Corner transfer matrix (CTM) for evaluating quantities



Treatment of long range ITE tensor by decomposition



Predefined Models

Spin system

$$\mathcal{H} = \sum_{ij}\left[\sum_{\alpha=x,y,z} J_{ij}^{\alpha} S_i^{\alpha} S_j^{\alpha} + B_{ij}\left(\vec{S}_i \cdot \vec{S}_j\right)^2\right] - \sum_i \left[\sum_{\alpha=x,y,z} h^{\alpha} S_i^{\alpha} + D\left(S_i^z\right)^2\right]$$

Bosonic system

$$\mathcal{H} = \sum_{ij}\left[-t_{ij}\left(b_i^{\dagger} b_j + b_j^{\dagger} b_i\right) + V_{ij} n_i n_j\right] + \sum_i \left[U\frac{n_i(n_i-1)}{2} - \mu n_i\right]$$

Predifined Lattices

Square lattice, Triangular lattice, Honeycomb lattice, Kagome lattice

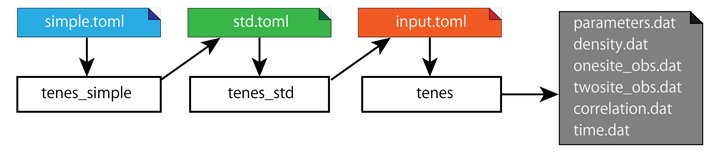
Available quantities

Onesite observables, e.g. $\langle S_i^z \rangle$ Twosite observables, e.g. $\langle S_i^z S_j^z \rangle$
Bond Hamiltonian $\langle \mathcal{H}_{ij} \rangle$ Correlation function

Download and Install

```
git clone https://github.com/issp-center-dev/TeNeS
cd TeNeS
mkdir build && cd build
cmake ../ -DCMAKE_INSTALL_PREFIX=$HOME/tenes
make install
```

Simple usage



Transverse field Ising model on a square lattice $\mathcal{H} = -\sum_{\langle ij\rangle} S_i^z S_j^z - \Gamma \sum_i S_i^x$

Prepare an input file, **simple.toml**, as follows

```
[parameter]
[parameter.simple_update]  # simple update
num_step = 1000            # number of steps
tau = 0.01                 # imaginary time slice
[parameter.full_update]    # full update
num_step = 100
tau = 0.01
[parameter.ctm]            # corner transfer matrix
dimension = 10             # bond dimension (chi)

[model]                    # Model parameter
type = "spin"              # "spin" or "boson"
S = 0.5                    # size of spin
Jz = -1.0                  # coupling constant
Jx = 0.0                   # ferromagnetic if negative
Jy = 0.0
hx = 3.0                   # magnetic field

[lattice]                  # Lattice parameter
type = "square"            # "square", "triangular", ...
L = 2                      # size of unitcell
W = 2                      # size of unitcell
virtual_dim = 2            # bond dimension
initial = "ferro"          # initial state
```

Convert simple.toml into **input.toml**, input of the main program

```
$HOME/tenes/bin/tenes_simple simple.toml   <-- make bond Ham.
$HOME/tenes/bin/tenes_std std.toml         <-- make ITE tensors
```

Invoke the main program **tenes**

```
mpiexec -np 4 $HOME/tenes/bin/tenes input.toml
```

and obtain the short summary as follows

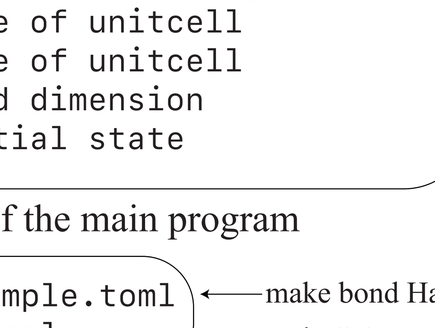
```
Onesite observables per site:
  Sz          = 1.24591818477e-09 8.39150297627e-21
  Sx          = 0.492501823783 8.07606608329e-20
  Sy          = -1.29807181831e-14 -7.12159178454e-17
Twosite observables per site:
  hamiltonian = -1.52140970487 -2.53451589044e-19
  SzSz        = 0.0439042335207 7.17040927922e-21
  SxSx        = 0.488692180826 8.34214740471e-20
  SySy        = -0.0406962922941 -3.54552009398e-21
```

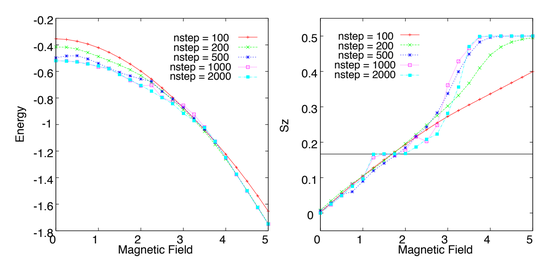
Detailed results are saved into the **output** directory.

Examples

Bond dimension (D) dependence of the energy of S=1/2 antiferromagnetic Heisenberg (AFH) model on a square lattice

unitcell: $L = W = 2$
dim. of CTM bond: $\chi = D^2$





unitcell: $L = W = 3$
bond dimension: $D = 2$
dim. of CTM bond: $\chi = 10$

Magnetization process and 1/3 plateau of S=1/2 AFH model on a triangular lattice

More example are available in the sample directory!

Contact

```
GitHub: https://github.com/issp-center-dev/TeNeS
e-mail: tenes-dev@issp.u-tokyo.ac.jp
```

Acknowledgement

TeNeS was supported by MEXT as "Exploratory Challenge on Post-K computer" (Frontiers of Basic Science: Challenging the Limits) and "Priority Issue on Post-K computer" (Creation of New Functional Devices and High-Performance Materials to Support Next-Generation Industries). We also would also like to express our thanks for the support of the "Project for advancement of software usability in materials science" of The Institute for Solid State Physics, The University of Tokyo, for the development of TeNeS.