Implementation of density matrix renormalization group (DMRG) for zigzag graphene nanoribons (ZGNR)

Le Hoang Anh
Department of Physics, Korea University, Seoul 02855, Korea

In this note, I am going to present what I have learnt so far from DMRG algorithm. This body of this note consists of two main parts: theoretical review of DMEG algorithm and personal experience of performing DMRG calculation on zigzag graphene nanoribbon with Julia language [1]. The last section of this note is devoted to answer questions of Professor Yang.

Version: Friday 22nd April, 2022, 22:28

I. THEORETICAL REVIEW OF DMRG

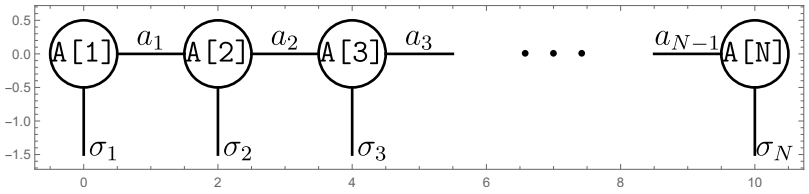
I.1. Matrix product state (MPS), why can MPS effectively represent many-body wavefunctions?

MPS can represent a variety of physical systems. However, we narrow down the scope of this note to spinful electrons. Here we consider a system with N electron sites whose many-body wavefunction can be represented as

$$|\Psi\rangle = \sum_{\sigma_1\sigma_2\ldots\sigma_N} \psi_{\sigma_1\sigma_2\ldots\sigma_N} |\sigma_1\sigma_2\ldots\sigma_N\rangle , \tag{1}$$

where $|\sigma_i\rangle$ denotes four occupation states at site i: $\{|\sigma_i\rangle\} = \{|0\rangle, |\uparrow\rangle, |\downarrow\rangle, |\uparrow\downarrow\rangle\}$. $\psi_{\sigma_1\sigma_2\ldots\sigma_N}$ is called *generic* representation and it grows exponentially with the number of sites N. The exponential wall can be avoided by representing $|\Psi\rangle$ in MPS form as follows

$$|\Psi\rangle = \sum_{\sigma_1\sigma_2\ldots\sigma_N} \sum_{a_1a_2\ldots a_N} A^{\sigma_1}_{a_1} A^{\sigma_2}_{a_1a_2} \ldots A^{\sigma_N}_{a_{N-1}} |\sigma_1\sigma_2\ldots\sigma_N\rangle . \tag{2}$$



The generic form is essentially singular-value-decomposed (SVDed) into a multiplication of matrices where `A[i]` is associated to the i^{th} electron site (this probably explains the name *matrix product state*). In the MPS diagram above, $a_1, a_2, \ldots a_{N-1}$ denote bond indices which are summed over in the multiplication while $\sigma_1, \sigma_2, \ldots, \sigma_N$ denote physical indices encoding the four occupation states. Notice that except for the matrices associated to the first and last sites, i.e., `A[1]` and `A[2]` are matrices, others are indeed ranked-3 tensors. However, they can be reshaped into a matrix and thus, can perform the matrix multiplication as we have mentioned.

If dimensions of bond indices are allowed to be infinitely large, MPS could represent exactly any generic form many-body wavefunction. To obtain an approximated representation, we truncate bond's dimensions, i.e., we only keep states corresponding to a certain number of largest singular values of the SVDs. Here we face a natural question: *Does the approximated MPS effectively represent* $|\Psi\rangle$*?* The answer is yes, due to *area law*. The area law states that for certain states, the entanglement rank across any bisection is bounded. Such entanglement can be quantified by 0-Rényi entropy

$$S_0 = \log \mathrm{Tr}\left(\sum_i \lambda_i\right), \tag{3}$$

with λ_i being the Schmidt coefficients. (The term Schmidt decomposition is used for quantum density states, its classical counterpart is SVD of matrix representation of the quantum states. Hence, Schmidt coefficients are equivalent to singular values.) Notice that larger Schmidt coefficients contributes more to the entropy. Therefore, one can truncate the bond dimensions of MPS, in other words, remove the smallest Schmidt coefficients across every bond, but the MPS still effectively represents the many-body state $|\Psi\rangle$ (see Refs. [2–4]).

I.2. Matrix product operator

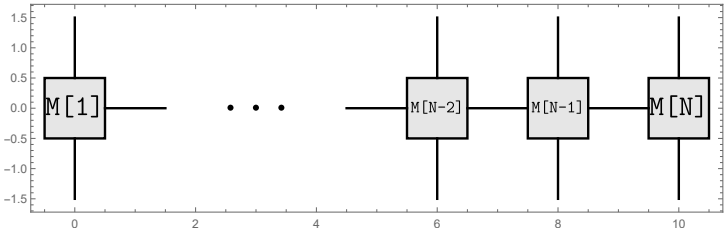
MPO formalism can be represented in a similar fashion to MPS in the previous subsection. For a system consisting N spinful electrons, a general operator can be written as

$$\hat{M} = \sum_{[\sigma,\sigma']} M^{\sigma_1'\sigma_2'\ldots\sigma_N'}_{\sigma_1\sigma_2\ldots\sigma_N} \left|\sigma_1'\sigma_2'\ldots\sigma_N'\right\rangle \left\langle\sigma_1\sigma_2\ldots\sigma_N\right|, \tag{4}$$

where the generic term could be decomposed into

$$M^{\sigma_1'\sigma_2'\ldots\sigma_N'}_{\sigma_1\sigma_2\ldots\sigma_N} = M^{\sigma_1\sigma_1'}_{m_1} M^{\sigma_2\sigma_2'}_{m_1 m_2} \cdots M^{\sigma_N\sigma_N'}_{m_{N-1}}. \tag{5}$$

The schematic illustrating MPO is drawn below.



I.3. Tensor network in the presence of symmetry

Because ZGNR with Hubbard model posses U(1) symmetry or the electron-number conservation, it is worth to mention that by judiciously exploiting the symmetry, DMRG algorithm could converge faster and require less resources [5]. In this section, we are going to briefly explain why DMRG with symmetry could have such advantage. (I will fill out this subsection in later version.)

I.4. DMRG algorithm

The term "renormalization group" in DMRG has a historical reason: the algorithm was indeed adapted from renormalization procedure. Briefly saying, a renormalization procedure starts from a small subsystem, then solves eigenvalue problem and truncates the initial Hamiltonian with a certain number of lowest-lying eigenstates. After each truncating step, the procedure restarts with a doubled-size subsystem and repeats until it reaches full size of system. The aforementioned procedure grows exponentially.

In Ref. [6] S. White proposed DMRG with a linear growth. The utmost goal of DMRG algorithm is to minimize energy of DMRG variational wavefunction. The algorithm can be outlined as follows [7].

(i) To start the DMRG procedure, we assign an initial MPS to the system. For two or higher dimensional systems, one needs to map it onto a 1D ones. A detailed example of this step will be written in subsection II.2. Notice that in case of using symmetry, the symmetry properties must be encoded in the initial MPS. For example, if we want to employ U(1) symmetry, the initial MPS should be declared with a well-defined number of electrons (or total magnetic S_z).

(ii) Sweeping process: the main part of DMRG algorithm consists of iteratively sweeping forward and backward through the 1D system chain, one complete going back and forth the chain is counted as one sweep. Each sweep solves ground-state problems in each site pair, performs SVD and keeps a certain number of states with largest singular values.

As an example, Fig. 1 describes the sweeping step at the right-boundary site `A[N]`. Notice that at each step, the system is divided in to left and right blocks which associate with left and right renormalized operators (in the figure, it is right renormalized operator `R[N-2]` updated by `R[N-1]` and the truncated MPS at site `B[N-1]`. They are also called complementary operators which are key elements to DMRG algorithm efficiently.

After initially declared number of sweeps is reached, the DMRG algorithm stops. In practice, we raise the number of sweeps until we find the system converged.

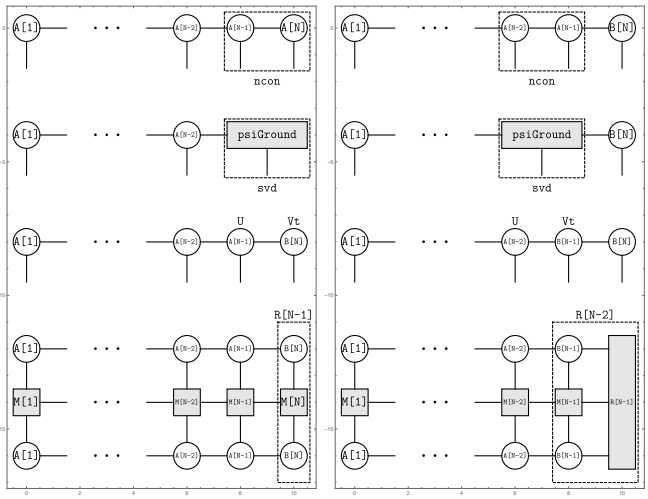


FIG. 1. Schematic diagram of a DMRG sweeping step.

II. PERSONAL EXPERIENCE OF DMRG CALCULATIONS ON ZGNR

II.1. Quantify how the truncation affects DMRG performance

Usually the DMRG algorithm is used in a pragmatic fashion: we verify reliability of DMRG results by examining them with different truncated bond dimensions, number of sweeps and test whether physical quantities of obtained ground state agree with the well-known results. Therefore, my set-up DMRG parameters closely follow the set-up of Professor Lee Huyn-Yong.

Ref. [8, 9] provide a theoretical background of DMRG truncation. To mimic the thermodynamic limit, DMRG embeds the system in a similar-size environment to form *superblock*. If the superblock state $|\Psi\rangle$ is pure, one can trace out environment to obtain system's state

$$\rho_S = \mathrm{Tr}_E \left|\Psi\right\rangle \left\langle\Psi\right| = \sum_{\alpha=1}^{\chi} w_\alpha \left|w_\alpha\right\rangle \left\langle w_\alpha\right|, \tag{6}$$

where w_αs stand for Schmidt coefficients ($\sum_\alpha w_\alpha = 1$), χ is the system's dimension and we assume the descending order $w_1 \geq w_2 \geq w_3 \geq \cdots$. Average value of local quantity $\hat{A}$ takes the following form

$$\langle \hat{A} \rangle = \sum_{\alpha}^{\chi} w_\alpha \left\langle w_\alpha\right| \hat{A} \left|w_\alpha\right\rangle. \tag{7}$$

If we truncate the number Schmidt coefficients from χ to χ_{tr}, the error of local quantity $\hat{A}$ is mathematically quantified as

$$\begin{aligned} \left| \langle \hat{A} \rangle_{\mathrm{tr}} - \langle \hat{A} \rangle \right| &= \left| \sum_{\alpha}^{\chi_{\mathrm{tr}}} w_\alpha \left\langle w_\alpha\right| \hat{A} \left|w_\alpha\right\rangle - \sum_{\alpha}^{\chi} w_\alpha \left\langle w_\alpha\right| \hat{A} \left|w_\alpha\right\rangle \right| \\ &\leq \epsilon_\rho ||\hat{A}||_{\max}, \end{aligned} \tag{8}$$

where the truncated weight is

$$\epsilon_\rho = 1 - \sum_{\alpha=1}^{\chi_{\mathrm{tr}}} w_\alpha. \tag{9}$$

Eq. 9 says that the faster eigenvalue spectrum decays, the less error the truncation makes. We observe that in Fig. 7 located in the appendix where Prof. Lee computed edge magnetization versus various values of χ_{tr} ($\chi_{\max}$ in the literature), the magnetization drops monotonously once truncated dimension surpasses 200, exhibiting more precise results is achieved with less truncation. That explains why $\chi_{\mathrm{tr}} = 1600$ was used throughout in his DMRG calculations.

II.2. Hubbard Hamiltonian of ZGNR and order of wrapping the lattice

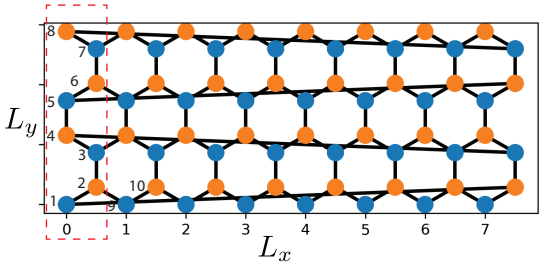


FIG. 2. The zigzag graphene nanoribbon with snake pattern and site indices are shown. The black lines indicate possible hopping elements between two connected sites. L_x and L_y respectively stand for the number of columns and rows (or the ribbon length and width). In the above figure, the first column lies within the red dashed box, and there are 8 rows and 8 columns.

The Hubbard-model Hamiltonian associated with the nanoribbon in Fig. 2 reads

$$H = -t \sum_{\langle ij \rangle \sigma} c^{\dagger}_{i,\sigma} c_{j,\sigma} + \sum_{i,\sigma} V_i c^{\dagger}_{i,\sigma} c_{i,\sigma} + U \sum_{i} n_{i,\uparrow} n_{i,\downarrow}, \tag{10}$$

with $n_{i,\sigma} \equiv c^{\dagger}_{i,\sigma} c_{i,\sigma}$ is the number operator of spin-σ electron, U is the on-site interacting strength, and V_i indicates the disorder strength at site i. Hopping terms between neighboring sites and the periodic boundary condition along L_x are represented by black solid lines.

The system is obviously 2D, to represent it in MPS and MPO formalism one needs to map it into a (quasi) 1D system. In tensor network language we call the mapping as wrapping the lattice. The order of wrapping illustrated in Fig. 2 is called *snake type*. There exists other orders like *Fstyle*, *snake Fstyle* and it is encouraged to try DMRG calculation with different wrapping orders [10]. (The DMRG results here are computed with snake style only in the spirit of reproducing Prof. Lee Huyn-Yong's calculation.)

II.3. Influence of initial states on DMRG calculation

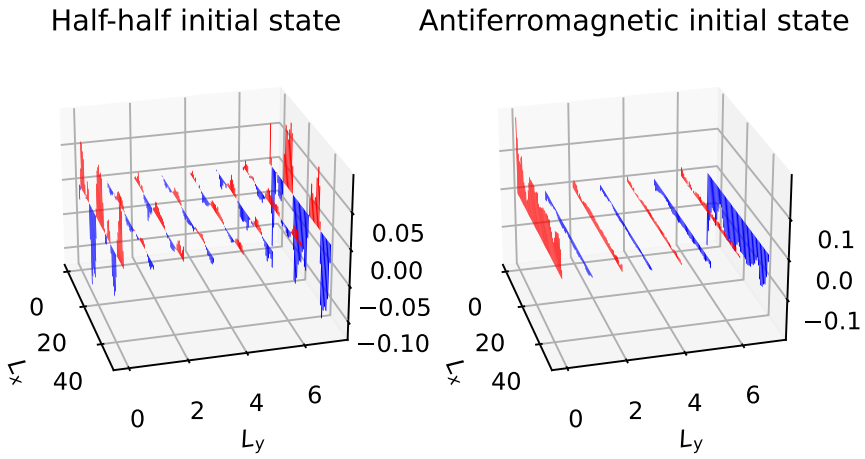


FIG. 3. (Disorder, half-filling case) DMRG parameters for two cases are set for the same number of sweeps, maximum bond indices and noise terms ($n_{\text{sweeps}} = 5, \chi_{\max} = 1600$). Here the system size is 50 by 8, two cases share the same Hamiltonian including disorder term ($U = t, \Gamma = 0.5t, n_{\text{imp=1}}$).

DMRG algorithm is a class of variational method, hence one expects that the algorithm will converge faster if we set an initial state with properties close to ones of ground state.

Fig. 3 illustrates the influence of initial state on DMRG. The parameters used for the two DRMG calculations are identical except for the initial states. The one with half-half initial state, i.e., $\langle n_\uparrow \rangle$ and $\langle n_\downarrow \rangle$ are randomly either 0.499 and 0.501, gives more accurate result. Also, the computation time for half-half initial state is 3 times lower than one for antiferromagnetic state.

II.4. Reported results

A DMRG calculation result with system size (50, 8) is reported in Fig. 4.

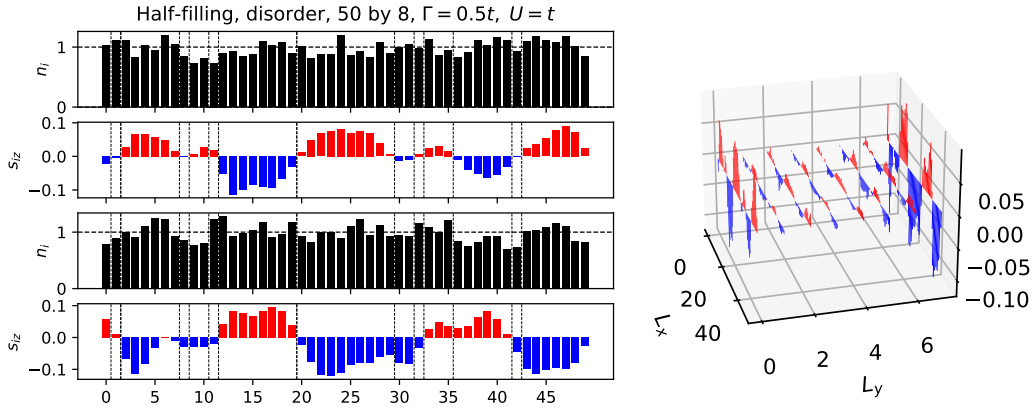


FIG. 4. $U = t, \Gamma = 0.5t, n_{\text{imp=1}}, n_{\text{sweeps}} = 5, \chi_{\max} = 1600$

Fig. 5 reports the disorder-induced site occupation and bipartite entanglement entropy for system size (12, 4) at doped case.

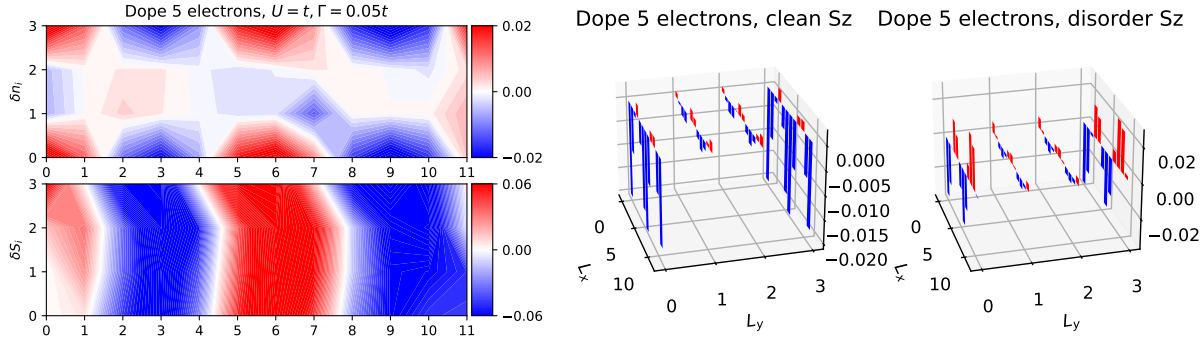


FIG. 5. $U = t, \Gamma = 0.05t, n_{\text{imp=1}}, n_{\text{sweeps}} = 5, \chi_{\max} = 1600$

III. ANSWERS TO PROFESSOR YANG'S QUESTIONS

(i) Investigate the role of initial states

The authors of DMRG-calculationi library [1] recommend to use initial states with properties close to the ground state. From my experience, the DMRG calculation was significantly improved as I changed from initial antiferromagnetic state to initial half-half state. The detailed analysis is mentioned in previous section II.3.

(ii) Maximal bond dimension and number of sweeps

As analyzed in subsection II.1, here we set maximal bond dimension $\chi_{\max}$ equal 1600 throughout the calculations as Prof. Lee did.

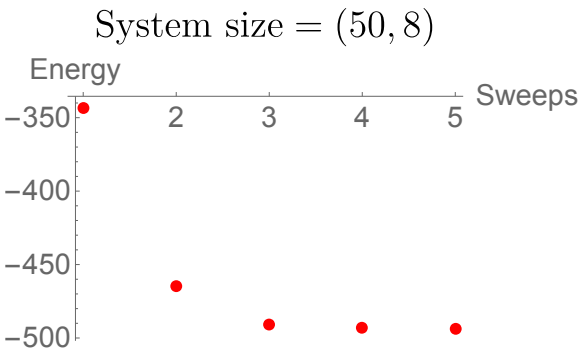


FIG. 6. Solved lowest energy after each sweep plot.

According to Fig. 6, the solved lowest energy barely changes after third sweep, hence the DMRG calculation for relevant system with 5 weeps could achieve a moderately accurate result. For a highly accurate result, one could set 10 sweeps. However, for such 10 sweeps, my personal experience is that it may take a usual CPU more than a week computing lowest energy for a system size (50, 8).

(iii) Are results with width $L_y = 4$ qualitatively different from ones with width $L_y = 8$?

Fig. 7 reports DMRG calculation with system size $(50, 4)$. Compared with system size $(50, 8)$ reported in Fig. 4, edge magnetization changes signs less times and thus, the result qualitatively differs.

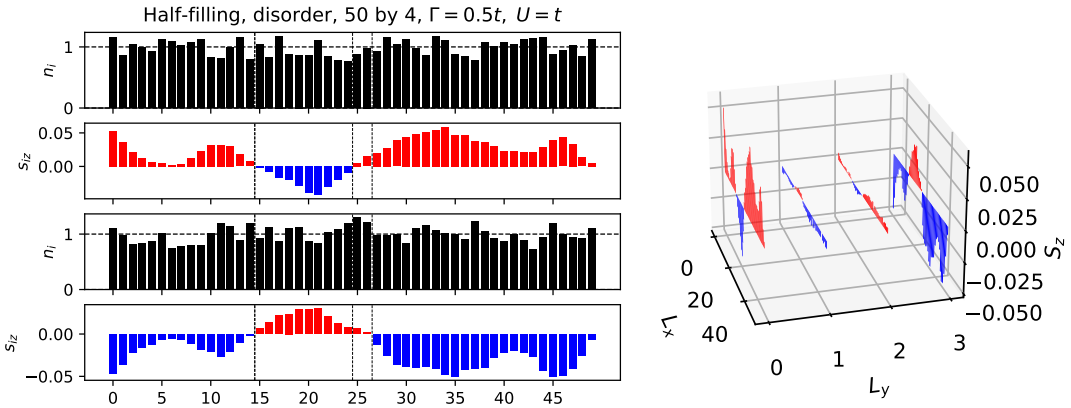


FIG. 7. $U = t, \Gamma = 0.5t, n_{\text{imp}=1}, n_{\text{sweeps}} = 5, \chi_{\max} = 1600$

REFERENCES

[1] M. Fishman, S. R. White, and E. M. Stoudenmire, The ITensor software library for tensor network calculations (2020), arXiv:2007.14822.

[2] T. E. Baker and M. P. Thompson, Build your own tensor network library: Dmrjulia i. basic library for the density matrix renormalization group, arXiv preprint arXiv:2109.03120 (2021).

[3] J. I. Cirac, D. Pérez-García, N. Schuch, and F. Verstraete, Matrix product states and projected entangled pair states: Concepts, symmetries, theorems, Rev. Mod. Phys. **93**, 045003 (2021).

[4] J. C. Bridgeman and C. T. Chubb, Hand-waving and interpretive dance: an introductory course on tensor networks, Journal of physics A: Mathematical and theoretical **50**, 223001 (2017).

[5] S. Singh, R. N. C. Pfeifer, and G. Vidal, Tensor network states and algorithms in the presence of a global u(1) symmetry, Phys. Rev. B **83**, 115125 (2011).

[6] S. R. White, Density matrix formulation for quantum renormalization groups, Phys. Rev. Lett. **69**, 2863 (1992).

[7] G. K.-L. Chan, A. Keselman, N. Nakatani, Z. Li, and S. R. White, Matrix product operators, matrix product states, and ab initio density matrix renormalization group algorithms, The Journal of Chemical Physics **145**, 014102 (2016), https://doi.org/10.1063/1.4955108.
[8] U. Schollwöck, The density-matrix renormalization group, Rev. Mod. Phys. **77**, 259 (2005).
[9] U. Schollwöck, The density-matrix renormalization group in the age of matrix product states, Annals of Physics **326**, 96 (2011), january 2011 Special Issue.
[10] J. Hauschild and F. Pollmann, Efficient numerical simulations with Tensor Networks: Tensor Network Python (TeNPy), SciPost Phys. Lect. Notes , 5 (2018), code available from `https://github.com/tenpy/tenpy`, arXiv:1805.00055.