

TOPICAL REVIEW • OPEN ACCESS

Tensor renormalization group for fermions

To cite this article: Shinichiro Akiyama *et al* 2024 *J. Phys.: Condens. Matter* **36** 343002

View the [article online](#) for updates and enhancements.

You may also like

- [Grassmann corner transfer-matrix renormalization group approach to one-dimensional fermionic models](#)
Jian-Gang Kong, , Zhi-Yuan Xie et al.
- [Variational Corner Transfer Matrix Renormalization Group Method for Classical Statistical Models](#)
X. F. Liu, , Y. F. Fu et al.
- [Building projected entangled pair states with a local gauge symmetry](#)
Erez Zohar and Michele Burrello

Topical Review

Tensor renormalization group for fermions

Shinichiro Akiyama^{1,2} , Yannick Meurice^{3,*}  and Ryo Sakai⁴ 

¹ Center for Computational Sciences, University of Tsukuba, Tsukuba, Ibaraki 305-8577, Japan

² Graduate School of Science, The University of Tokyo, Bunkyo-ku, Tokyo 113-0033, Japan

³ Department of Physics and Astronomy, The University of Iowa, Iowa City, IA 52242, United States of America

⁴ Jij Inc., Bunkyo-ku, Tokyo 113-0031, Japan

E-mail: yannick-meurice@uiowa.edu and akiyama@ccs.tsukuba.ac.jp

Received 6 February 2024, revised 2 April 2024

Accepted for publication 3 May 2024

Published 28 May 2024



Abstract

We review the basic ideas of the tensor renormalization group method and show how they can be applied for lattice field theory models involving relativistic fermions and Grassmann variables in arbitrary dimensions. We discuss recent progress for entanglement filtering, loop optimization, bond-weighting techniques and matrix product decompositions for Grassmann tensor networks. The new methods are tested with two-dimensional Wilson–Majorana fermions and multi-flavor Gross–Neveu models. We show that the methods can also be applied to the fermionic Hubbard model in 1+1 and 2+1 dimensions.

Keywords: tensor networks, lattice gauge theory, relativistic lattice fermions, Fermi Hubbard model, Grassmann path integrals, sign problems

Contents

1. Introduction	2	3.2. The Schwinger model	18
2. Formalism for the Grassmann TRG	3	4. Improved TRG methods for fermions	19
2.1. Grassmann tensor network representation	3	4.1. CDL structure on tensor network	19
2.2. Exact contraction	6	4.2. Removal of CDL from network	19
2.3. Model-independent notation	8	4.3. Bond-weighting technique	20
2.4. Extension to lattice gauge theories	9	4.4. Multilayered tensor network formulations for N_f -flavor fermions	22
2.5. Approximate contraction by the Levin–Nave TRG	11	5. Relativistic models with fermion interactions	23
2.6. Approximate contraction by the HOTRG	14	5.1. Gross–Neveu model	23
2.7. Practical remarks	16	5.2. QCD in the infinite-coupling limit	24
3. Examples of numerical calculations	17	5.3. NJL model	25
3.1. Wilson–Majorana fermions	17	5.4. $\mathcal{N} = 1$ Wess–Zumino model	26
		5.5. Non-abelian lattice gauge theories coupled to fermions	26
		6. The Hubbard model	26
		6.1. (1+1)-dimensional model	28
		6.2. (2+1)-dimensional model	28
		7. Conclusions	29
		Data availability statement	29
		Acknowledgment	29
		References	29

* Author to whom any correspondence should be addressed.



Original Content from this work may be used under the terms of the [Creative Commons Attribution 4.0 licence](https://creativecommons.org/licenses/by/4.0/). Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

1. Introduction

The renormalization group (RG) approach of lattice models has been crucial to identify their universal critical behavior, construct their phase diagram and understanding their continuum limits. The general idea [1–3] consists in integrating over some of the microscopic degrees of freedom in order to obtain an effective theory with a larger lattice spacing and repeat the procedure until one reaches a macroscopic size. This leads to an RG map connecting effective theories at increasing lattice spacing. The fixed points and relevant directions of the RG maps have universal properties such as similar critical exponents observed in very different microscopic setups (e.g. magnets and solids).

The generic properties of RG maps are very well-understood [2–4]. However, the numerical implementation of the partial integration over some microscopic degrees of freedom (coarse-graining) can be challenging. This requires to parameterize some ‘space of theories’ in terms of effective couplings and find numerical methods to calculate the ‘new’ couplings in terms of the ‘old’ couplings. This provides the RG map. Simplified RG maps such as various majority rules, bond moving [5, 6] approximate recursions [2], hierarchical approximations [7–9], local potential approximations [10–12], can be easily implemented numerically and demonstrate the generic validity of the RG approach. However, these approximations provide critical exponents which are different from the exponents of the approximated models and it is difficult to systematically improve such approximations.

The basic RG ideas were incorporated in variational algorithms designed to construct the ground state wavefunction of Hamiltonians for one-dimensional spatial lattices often called the density matrix RG method [13–15]. This led to the development of tensor network ansatzes (e.g. matrix product states (MPS)) for these wavefunctions. It became clear that the success of the method was linked to its ability to handle efficiently the entanglement entropy in one spatial dimension [14, 16–25]. The tensor technology was also used to reformulate and coarse-grain classical lattice models [26–34]. This new approach is called the tensor RG (TRG). The TRG approach allows us to interpret and quantitatively realize the traditional RG ideas in the language of the tensor network [20, 35–41].

Stimulated by a certain number of interdisciplinary workshops (see for instance, [42]), tensorial methods became of interest to the lattice gauge theory community. MPS were used for the Schwinger model [43–51], non-Abelian gauge theories [52–55], and the $O(3)$ nonlinear sigma model [56]. Tensor network techniques for lattice gauge theories are also discussed in [57–62] and reviewed in [63, 64]. At the same time, it was shown that character expansions used in the context of the strong coupling expansion [65, 66] could be used for TRG approaches of most models studied in lattice gauge theory [67–71]. This led to a new unified way to understand global, local, continuous, and discrete symmetries [72, 73] more generally to the development of tensor lattice field theory recently reviewed in [74].

Fermionic tensor network studies with applications for the Fermi–Hubbard model were discussed in [75, 76]. The application of the TRG method to fermionic systems relying on Grassmann numbers was proposed by Gu, Verstraete, and Wen in [30, 31] in the context of the tensor product states (TPS) [77, 78] and projected entangled pair states (PEPS) [79]. TPS and PEPS are known as variational wavefunctions that can efficiently represent the ground state of a gapped local Hamiltonian in higher dimensions. They extended TPS to fermionic systems and proposed to construct the TPS using the Grassmann variables based on the idea of the path-integral formalism of fermionic systems. This generic variational wavefunction is referred to as the Grassmann TPS (GTPS). The TRG, which explicitly includes the Grassmann variables, was devised as a method for approximate contraction of the Grassmann tensor network derived from the GTPS. Note that [30] showed that the fermionic PEPS (fPEPS) proposed in [80] can be classified as a special subclass of GTPS. fPEPS and GTPS can be considered as an origin of the current Grassmann TRG approach [30, 31, 80]. Note that several kinds of higher-dimensional tensor network ansatzes with variational methods, some of which are combined with the Monte Carlo method, have recently been applied to various lattice gauge theories [81–92]. For a set of lectures notes on the applications of PEPS in the context of lattice gauge theory see [93].

A practically important aspect of tensor network including the TRG is that in principle, the sign problem does not exist. Although Quantum Monte Carlo simulations are very efficient for bosons, they face the negative sign problem when applied to fermions [94]. Introducing auxiliary bosons, the Monte Carlo methods can simulate lattice fermion models but it is usually computationally challenging because the resulting bosonic theory becomes completely non-local. For the numerical approaches for lattice fermions based on the Monte Carlo method, see references [95–98] and references therein. In contrast, the tensor network approach allows us to deal with local fermionic actions because we can directly manipulate the Grassmann numbers. Moreover, if the system has translational invariance on a lattice, we can access its infinite-volume limit (zero-temperature limit) by using the TRG. As we will see, this advantage of the TRG has been demonstrated in actual numerical calculations for lattice fermions. Therefore, the tensor network approach has been a powerful candidate for investigating systems where the sign problem makes Quantum Monte Carlo simulation difficult to use in the first place.

In this topical review, we discuss recent TRG applications for fermionic models. The starting point is a path integral involving Grassmann variables. We will consider models in Euclidean spacetime and the statistical weights in the path integral will have the generic form e^{-S} for some classical action S . The first step of the tensorial approach consists in expanding the exponential of the non-local parts of S (attached to links or plaquettes of the lattice) in terms of discrete tensor indices and then perform the integration over the original variables. With this procedure, the path integral becomes a sum

of product of local tensors with their indices contracted. For bosonic variables, the tensors are just ordinary functions of the coupling that can be collected in a straightforward manner. However, for Grassmann variables, one needs to keep track of the signs resulting from various orderings. As we will show in section 2, this can be handled by introducing auxiliary Grassmann variables which will be incorporated in the tensors. We then proceed to coarse-grain the reformulation of the path integral by performing approximately partial contractions following the early TRG method of Levin–Nave (illustrated in figure 5) and their higher-order TRG (HOTRG) version (illustrated in figure 7). In section 3, we review simple examples of numerical calculations for free Wilson–Majorana fermions (equivalent to the Ising model) and 1+1 QED (the Schwinger model).

An important aspect of the TRG coarse-graining is that it can in principle be performed exactly [34, 67]. However, in practice, the computational cost still scales exponentially with the size of the system because despite the partial integration over some of the microscopic degrees of freedom, effective tensors with more indices are generated. The scaling is less severe, but nevertheless exponential. For this reason and also in order to get RG maps relating same size tensors, truncations are necessary. It has been argued that the inverse of the truncation size can be considered as a relevant direction which can interfere with the study of fixed points [99]. It has also been known [27, 28] that short-range entanglement can remain present during TRG iterations and generate unphysical fixed points. Improvements of this situation for fermionic theories are discussed in section 4. We introduce the corner double line (CDL) fixed point and then discuss methods to remove them. This includes the tensor network renormalization (TNR) [20], the loop-TNR [100], and the gilt-TNR [101] algorithm.

Applications for relativistic models are presented in section 5 which covers the Gross–Neveu model [102, 103], QCD at infinite coupling [104], the Nambu–Jona-Lasinio (NJL) model [105], the $\mathcal{N} = 1$ Wess–Zumino model [106] and non-abelian gauge theories with fermion [107]. Finally, the Fermi–Hubbard model which has been extensively studied in condensed matter community [108–114] is considered in section 6 following the approach of [115] in 1+1 dimensions and [116] in 2+1 dimensions.

2. Formalism for the Grassmann TRG

We begin by explaining how to express fermionic path integrals in the language of tensor networks. Because of the nilpotency of the Grassmann variables, they can be straightforwardly rewritten with finite-dimensional tensors. We show how to construct the exact contraction between these fundamental tensors, which is necessary to reproduce original path integrals. However, the exact contraction is usually prohibited in practice because it requires exponentially large computational resources. One of the promising approaches is the TRG which carries out these contractions approximately. Although the TRG algorithms are often used to compute the partition

functions without the Grassmann variables, we will see that any TRG algorithm can be applied to fermionic path integrals.

2.1. Grassmann tensor network representation

For the time being in this section, we use the following simple quadratic model to explain how to derive the Grassmann tensor network representation,

$$S = -t \sum_{n \in \Lambda} \sum_{\nu=1}^2 [\bar{\psi}(n + \hat{\nu}) \psi(n) + \bar{\psi}(n) \psi(n + \hat{\nu})] + m \sum_n \bar{\psi}(n) \psi(n), \quad (1)$$

where we assume that $\psi(n)$ and $\bar{\psi}(n)$ are single-component Grassmann fields for simplicity. We consider the model on a two-dimensional square lattice Λ with periodic boundary conditions. The path integral on the lattice is defined via

$$Z = \int \prod_n d\psi(n) d\bar{\psi}(n) e^{-S}. \quad (2)$$

Our goal is to rewrite equation (2) introducing local tensors. There are multiple ways to define these local tensors so that the tensor network representation is in general not unique. This situation is the same as in constructing tensor network representations for theories that do not include fermions.

One methodology was given by Shimizu and Kuramashi [117] and subsequently refined by Takeda and Yoshimura by explicitly introducing auxiliary Grassmann variables [102]. Here, we review the formalism in [102]. Firstly, the hopping terms are decomposed introducing auxiliary Grassmann variables as

$$\begin{aligned} e^{t\bar{\psi}(n+\hat{\nu})\psi(n)} &= \sum_{i_\nu(n)=0}^1 \left(\int \sqrt{t} \bar{\psi}(n+\hat{\nu}) d\bar{\Phi}_\nu(n+\hat{\nu}) \right)^{i_\nu(n)} \\ &\quad \times \left(\int \sqrt{t} \psi(n) d\Phi_\nu(n) \right)^{i_\nu(n)} (\bar{\Phi}_\nu(n+\hat{\nu}) \Phi_\nu(n))^{i_\nu(n)}, \end{aligned} \quad (3)$$

$$\begin{aligned} e^{t\bar{\psi}(n)\psi(n+\hat{\nu})} &= \sum_{j_\nu(n)=0}^1 \left(\int \sqrt{t} \bar{\psi}(n) d\bar{\Psi}_\nu(n) \right)^{j_\nu(n)} \\ &\quad \times \left(\int \sqrt{t} \psi(n+\hat{\nu}) d\Psi_\nu(n+\hat{\nu}) \right)^{j_\nu(n)} \\ &\quad \times (\bar{\Psi}_\nu(n) \Psi_\nu(n+\hat{\nu}))^{j_\nu(n)}, \end{aligned} \quad (4)$$

where Ψ_ν , $\bar{\Psi}_\nu$, Φ_ν , and $\bar{\Phi}_\nu$ are single-component Grassmann variables along the ν -directional link. The bit $i_\nu(n)$ ($j_\nu(n)$) labels the Taylor expansion of the forward (backward) hopping term on the link $(n, n + \hat{\nu})$. These bits are nothing but the occupation numbers. We use capital Greek letters for the

auxiliary Grassmann variables ($\Psi_\nu, \bar{\Psi}_\nu$, etc) as opposed to the original microscopic Grassman variables which are lowercase ($\psi, \bar{\psi}$). In the rest of this section, in order to have more compact notations, we will drop the $\int$'s in equations (3) and (4). In other words, when $i_\nu(n) = 1$ or $j_\nu(n) = 1$, the differentials of the auxiliary Grassmann variables such as $d\Psi_\nu$, should be understood as $\int d\Psi_\nu$. In the right-hand sides of equations (3) and (4), the original fields living on different sites are decomposed into different Grassmann-even pairs. Therefore, $\psi(n)$ and $\bar{\psi}(n)$ in equation (2) can be easily integrated at each lattice site n independently. At each lattice site n , we consider the following integral

$$\begin{aligned} & \mathcal{T}_{n;(i_1 j_1)(i_2 j_2)(i'_1 j'_1)(i'_2 j'_2)} \\ &= \int d\psi d\bar{\psi} e^{-m\bar{\psi}\psi} \prod_\nu (\sqrt{t}\psi d\Phi_\nu(n))^{i_\nu} (\sqrt{t}\bar{\psi} d\bar{\Psi}_\nu(n))^{j_\nu} \\ & \quad \times (\sqrt{t}\bar{\psi} d\bar{\Phi}_\nu(n))^{i'_\nu} (\sqrt{t}\psi d\Psi_\nu(n))^{j'_\nu} \\ & \quad \times (\bar{\Phi}_\nu(n+\hat{\nu})\Phi_\nu(n))^{i_\nu} (\bar{\Psi}_\nu(n)\Psi_\nu(n+\hat{\nu}))^{j_\nu}, \end{aligned} \quad (5)$$

where we have introduced several shorthand notations such as $i_\nu := i_\nu(n)$, $j_\nu := j_\nu(n)$, $i'_\nu := i_\nu(n-\hat{\nu})$, and $j'_\nu := j_\nu(n-\hat{\nu})$. We regard equation (5) as a fundamental tensor describing the path integral in equation (2). Equation (5) can be written as

$$\begin{aligned} & \mathcal{T}_{n;(i_1 j_1)(i_2 j_2)(i'_1 j'_1)(i'_2 j'_2)} \\ &= T_{n;(i_1 j_1)(i_2 j_2)(i'_1 j'_1)(i'_2 j'_2)} \mathcal{G}_{n;(i_1 j_1)(i_2 j_2)(i'_1 j'_1)(i'_2 j'_2)}, \end{aligned} \quad (6)$$

with

$$\begin{aligned} & T_{n;(i_1 j_1)(i_2 j_2)(i'_1 j'_1)(i'_2 j'_2)} \\ &= \sqrt{t}^{\sum_\nu (i_\nu + j_\nu + i'_\nu + j'_\nu)} (-1)^{j_1(i_2 + j'_1 + j'_2) + j_2(j'_1 + j'_2) + i'_1 j'_2} \\ & \quad \times [-m\delta_{i_1 + i_2 + j'_1 + j'_2, 0} \delta_{i'_1 + i'_2 + j_1 + j_2, 0} \\ & \quad + \delta_{i_1 + i_2 + j'_1 + j'_2, 1} \delta_{i'_1 + i'_2 + j_1 + j_2, 1}], \end{aligned} \quad (7)$$

$$\begin{aligned} & \mathcal{G}_{n;(i_1 j_1)(i_2 j_2)(i'_1 j'_1)(i'_2 j'_2)} \\ &= d\Phi_1(n)^{i_1} d\bar{\Psi}_1(n)^{j_1} d\Phi_2(n)^{i_2} d\bar{\Psi}_2(n)^{j_2} d\Psi_1(n)^{i'_1} \\ & \quad \times d\bar{\Phi}_1(n)^{i'_1} d\Psi_2(n)^{j'_2} d\bar{\Phi}_2(n)^{i'_2} \\ & \quad \times \prod_\nu (\bar{\Phi}_\nu(n+\hat{\nu})\Phi_\nu(n))^{i_\nu} (\bar{\Psi}_\nu(n)\Psi_\nu(n+\hat{\nu}))^{j_\nu}. \end{aligned} \quad (8)$$

The Kronecker deltas in equation (7) imply that

$$i_1 + j_1 + i_2 + j_2 + i'_1 + j'_1 + i'_2 + j'_2 \pmod{2} = 0. \quad (9)$$

In other words, the fundamental tensor $\mathcal{T}_n$ in equation (6) is always Grassmann-even. Note that the Kronecker deltas in equation (7) are *not* mod 2 because one may not have more than one power of the Grassman numbers. Following [102], we call T_n in equation (7) as the bosonic part of $\mathcal{T}_n$ and $\mathcal{G}_n$ as the Grassmann part. Note the order of the Grassmann measures in equation (8). Although $\Phi_\nu(n)^{i_\nu}$ and $\bar{\Psi}_\nu(n)^{j_\nu}$ ($\nu = 1, 2$) can be integrated within equation (8), the integration shall not be performed at this stage. The graphical representation of

equation (6) is provided in figure 1 (notice the left-right asymmetry). Z in equation (2) is reproduced by summing over all bits and integrating over all auxiliary Grassmann variables. This situation is symbolically expressed as

$$Z = \sum_{\{i_1, j_1, i_2, j_2\}} \int \prod_n \mathcal{T}_n, \quad (10)$$

where

$$\sum_{\{i_1, j_1, i_2, j_2\}} = \prod_{n, \nu} \sum_{i_\nu(n)=0}^1 \sum_{j_\nu(n)=0}^1, \quad (11)$$

and $\int$'s are over the auxiliary Grassmann variables, which is implicit in equation (8).

This kind of Grassmann tensor network formulation has been widely applied: the Schwinger model [117–119]⁵, the Gross–Neveu model [102, 120], free Wilson fermions [121, 122], the $\mathcal{N} = 1$ Wess–Zumino model [106], the NJL model [105], Wilson–Majorana fermions [120], infinite-coupling QCD [104], and $SU(2)$ lattice gauge theory with reduced staggered fermions [107].

There are alternative ways to formulate the tensor network with Grassmann variables. Meurice introduced a multilinear combination of Grassman variables to define a fundamental tensor [123] and Bao summarized the contraction, decomposition, and conjugation for the tensors associated with the Grassmann variables [124]. Akiyama and Kadoh then gave a general methodology to derive the tensor network representation for fermionic path integrals based on the concrete definition of the Grassmann tensor [125]. Here, we follow the formalism provided in [125].

They also decompose the hopping terms but use

$$\begin{aligned} e^{t\bar{\psi}(n+\hat{\nu})\psi(n)} &= \int d\bar{\Phi}_\nu(n) d\Phi_\nu(n) e^{-\bar{\Phi}_\nu(n)\Phi_\nu(n)} \\ & \quad \times e^{\sqrt{t}\bar{\psi}(n+\hat{\nu})\bar{\Phi}_\nu(n)} e^{\sqrt{t}\psi(n)\Phi_\nu(n)}, \end{aligned} \quad (12)$$

$$\begin{aligned} e^{t\bar{\psi}(n)\psi(n+\hat{\nu})} &= \int d\bar{\Psi}_\nu(n) d\Psi_\nu(n) e^{-\bar{\Psi}_\nu(n)\Psi_\nu(n)} \\ & \quad \times e^{\sqrt{t}\bar{\psi}(n)\bar{\Psi}_\nu(n)} e^{-\sqrt{t}\psi(n+\hat{\nu})\bar{\Psi}_\nu(n)}, \end{aligned} \quad (13)$$

instead of equations (3) and (4). Again, $\Psi_\nu, \bar{\Psi}_\nu, \Phi_\nu$, and $\bar{\Phi}_\nu$ are the single-component Grassmann variables. We are now ready to carry out the integral over $\psi(n)$ and $\bar{\psi}(n)$ in equation (2) at each site n independently as before. The fundamental tensor is defined via

$$\begin{aligned} \mathcal{T}_n &= \int d\psi d\bar{\psi} e^{-m\bar{\psi}\psi} \prod_\nu e^{\sqrt{t}\bar{\psi}\bar{\Phi}_\nu(n-\hat{\nu})} \\ & \quad \times e^{\sqrt{t}\bar{\psi}\bar{\Psi}_\nu(n)} e^{\sqrt{t}\psi\Phi_\nu(n)} e^{-\sqrt{t}\psi\Psi_\nu(n-\hat{\nu})}. \end{aligned} \quad (14)$$

⁵ Although the previous study in [117] does not appear to introduce auxiliary Grassmann variables explicitly, it gives a formulation that is equivalent to introducing auxiliary variables. Consequently, their fundamental tensor in equation (31) in [117] has the same structure with equation (6).

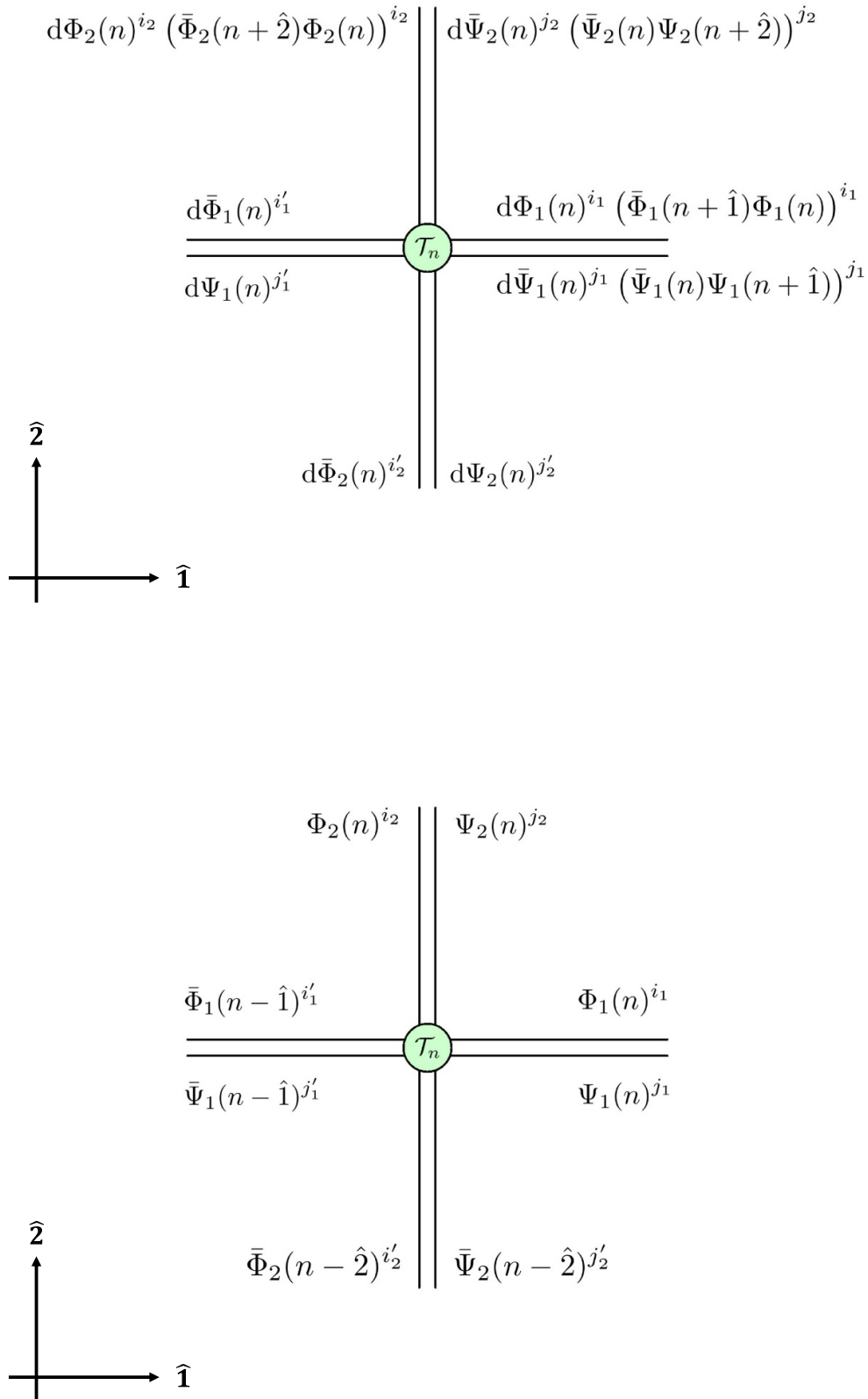


Figure 1. Graphical representation of the fundamental tensor $\mathcal{T}_n$. (Top) The fundamental tensor in equation (6). Each line denotes the single-component auxiliary Grassmann measure and the occupation number. Grassmann-even factors are associated with forward lines. (Bottom) The fundamental tensor in equation (15). Each line denotes the single-component auxiliary Grassmann variable.

Expanding the integrand, one obtains

$$\begin{aligned} \mathcal{T}_{n;\Phi_1\Psi_1\Phi_2\Psi_2\bar{\Psi}_1\bar{\Phi}_1\bar{\Psi}_2\bar{\Phi}_2} \\ = \left(\prod_{\nu} \sum_{i_{\nu}, j_{\nu}, i'_{\nu}, j'_{\nu}} \right) T_{n;(i_1j_1)(i_2j_2)(i'_1j'_1)(i'_2j'_2)} \\ \times \Phi_1^{i_1} \Psi_1^{j_1} \Phi_2^{i_2} \Psi_2^{j_2} \bar{\Psi}_1^{i'_1} \bar{\Phi}_1^{j'_1} \bar{\Psi}_2^{i'_2} \bar{\Phi}_2^{j'_2}, \end{aligned} \quad (15)$$

with

$$\begin{aligned} T_{n;(i_1j_1)(i_2j_2)(i'_1j'_1)(i'_2j'_2)} &= (-1)^{\sum_{\nu} j'_{\nu}} \sqrt{t}^{\sum_{\nu} (i_{\nu}+j_{\nu}+i'_{\nu}+j'_{\nu})} \\ &\times (-1)^{j_1(i_2+j'_1+j'_2)+j_2(j'_1+j'_2)+i'_1j'_2} \\ &\times [-m\delta_{i_1+i_2+j'_1+j'_2,0}\delta_{i'_1+i'_2+j_1+j_2,0} \\ &+ \delta_{i_1+i_2+j'_1+j'_2,1}\delta_{i'_1+i'_2+j_1+j_2,1}]. \end{aligned} \quad (16)$$

As for equations (7) and (16) shows explicitly that the fundamental tensor in equation (15) is Grassmann-even. Compared with the previous formulation based on [102], all the bits introduced by the Taylor expansion are summed within the fundamental tensor $\mathcal{T}_n$. Instead, we can identify the auxiliary Grassmann variables as the indices of $\mathcal{T}_n$. This is why we have introduced the notation as in the left-hand side of equation (15). Following [125], we refer $\mathcal{T}_n$ in equation (15) as the Grassmann tensor and T_n in equation (16) as the coefficient tensor of $\mathcal{T}_n$. The difference between the bosonic part in equation (7) and the coefficient tensor in equation (16) is just the sign factor $(-1)^{\sum_{\nu} j'_{\nu}}$ in equation (16), which originates from the negative sign in the last exponential factor in equation (13). The graphical expression of equation (15) is shown in figure 1. The path integral is reproduced from the fundamental tensor $\mathcal{T}_n$ in equation (15). Introducing the following abbreviation,

$$\int_{\bar{\Phi}, \Phi} = \int d\bar{\Phi} d\Phi e^{-\bar{\Phi}\Phi}, \quad (17)$$

and

$$\text{gTr}[\cdot] = \left(\prod_{\nu} \int_{\bar{\Phi}_{\nu}(n), \Phi_{\nu}(n)} \int_{\bar{\Psi}_{\nu}(n), \Psi_{\nu}(n)} \right) [\cdot], \quad (18)$$

Z in equation (2) is given by

$$Z = \text{gTr} \left[\prod_n \mathcal{T}_n \right]. \quad (19)$$

The symbol ‘gTr’ in equation (18) means the Grassmann tensor trace that is analogous to the tensor trace symbol ‘tTr’ common in the tensor network formulation in the spin systems.

This formulation has been applied recently for the free Wilson and staggered fermions [125], Hubbard models [115, 116], $N_f = 1, 2, 3$ Gross–Neveu model [103, 126], $\mathbb{Z}_n$ and $U(1)$ gauge theories with $N_f = 1, 2, 4$ Wilson fermions [127], and several public codes for the Grassmann TRG methods [128, 129].

2.2. Exact contraction

Let us figure out how to carry out the contraction between the fundamental tensors derived above. As an example, we consider the contraction between $\mathcal{T}_n$ and $\mathcal{T}_{n+\hat{1}}$, which reproduces the hopping terms on the link $(n, n+\hat{1})$. When we use the fundamental tensor given in equation (6), the contraction is defined by

$$\begin{aligned} \sum \int \mathcal{T}_{n+\hat{1}} \mathcal{T}_n \\ = \sum_{\alpha_1, \beta_1} T_{n+\hat{1};(i_1j_1)(i_2j_2)(\alpha_1\beta_1)(i'_2j'_2)} T_{n;(\alpha_1\beta_1)(k_2l_2)(k'_1l'_1)(k'_2l'_2)} \\ \times \int \mathcal{G}_{n+\hat{1};(i_1j_1)(i_2j_2)(\alpha_1\beta_1)(i'_2j'_2)} \mathcal{G}_{n;(\alpha_1\beta_1)(k_2l_2)(k'_1l'_1)(k'_2l'_2)}. \end{aligned} \quad (20)$$

Note that $\int$ on the right-hand side means the integration over the auxiliary Grassmann variables labeled by repeated Greek indices. Firstly, we integrate out $(\bar{\Phi}_1(n+\hat{1})\Phi_1(n))^{\alpha_1}$ and $(\bar{\Psi}_1(n)\Psi_1(n+\hat{1}))^{\beta_1}$ originating from $\mathcal{G}_n$, which results in

$$\begin{aligned} \int \mathcal{G}_{n+\hat{1};(i_1j_1)(i_2j_2)(\alpha_1\beta_1)(i'_2j'_2)} \mathcal{G}_{n;(\alpha_1\beta_1)(k_2l_2)(k'_1l'_1)(k'_2l'_2)} \\ = (-1)^{(\alpha_1+\beta_1)(i'_2+j'_2)+\alpha_1} \mathcal{Q}_{(i_1j_1)(i_2j_2)(i'_2j'_2)(k_2l_2)(k'_1l'_1)(k'_2l'_2)}, \end{aligned} \quad (21)$$

with

$$\begin{aligned} \mathcal{Q}_{(i_1j_1)(i_2j_2)(i'_2j'_2)(k_2l_2)(k'_1l'_1)(k'_2l'_2)} \\ = d\bar{\Phi}_1^{i_1} d\bar{\Phi}_1^{j_1} d\Psi_2^{i_2} d\Psi_2^{j_2} d\bar{\Psi}_2^{i'_2} d\bar{\Psi}_2^{j'_2} d\Psi_2^{k_2} d\Psi_2^{l_2} d\bar{\Psi}_1^{l'_1} d\bar{\Phi}_1^{k'_1} d\Psi_2^{l'_2} d\bar{\Phi}_2^{k'_2} \\ \times (\bar{\Phi}_1\Phi_1)^{i_1} (\bar{\Psi}_1\Psi_1)^{j_1} (\bar{\Phi}_2\Phi_2)^{i_2} (\bar{\Psi}_2\Psi_2)^{j_2} (\bar{\Phi}_2\Phi_2)^{k_2} (\bar{\Psi}_2\Psi_2)^{l_2}. \end{aligned} \quad (22)$$

We have omitted the site dependence in the auxiliary Grassmann variables in equation (22) because it can be read from the bit indices immediately. The sign in equation (21) is then taken into account by contracting the bosonic parts. As a result of equation (20), we obtain

$$\begin{aligned} \sum \int \mathcal{T}_{n+\hat{1}} \mathcal{T}_n &= M_{(i_1j_1)(i_2j_2)(i'_2j'_2)(k_2l_2)(k'_1l'_1)(k'_2l'_2)} \\ &\times \mathcal{Q}_{(i_1j_1)(i_2j_2)(i'_2j'_2)(k_2l_2)(k'_1l'_1)(k'_2l'_2)}, \end{aligned} \quad (23)$$

where

$$\begin{aligned} M_{(i_1j_1)(i_2j_2)(i'_2j'_2)(k_2l_2)(k'_1l'_1)(k'_2l'_2)} \\ = \sum_{\alpha_1, \beta_1} (-1)^{(\alpha_1+\beta_1)(i'_2+j'_2)+\alpha_1} T_{n+\hat{1};(i_1j_1)(i_2j_2)(\alpha_1\beta_1)(i'_2j'_2)} \\ \times T_{n;(\alpha_1\beta_1)(k_2l_2)(k'_1l'_1)(k'_2l'_2)}. \end{aligned} \quad (24)$$

It is important to realize that the auxiliary Grassmann variables should be integrated before the bosonic parts are contracted when we use the fundamental tensor given in equation (6). Figure 2 illustrates the exact contraction in equation (20),

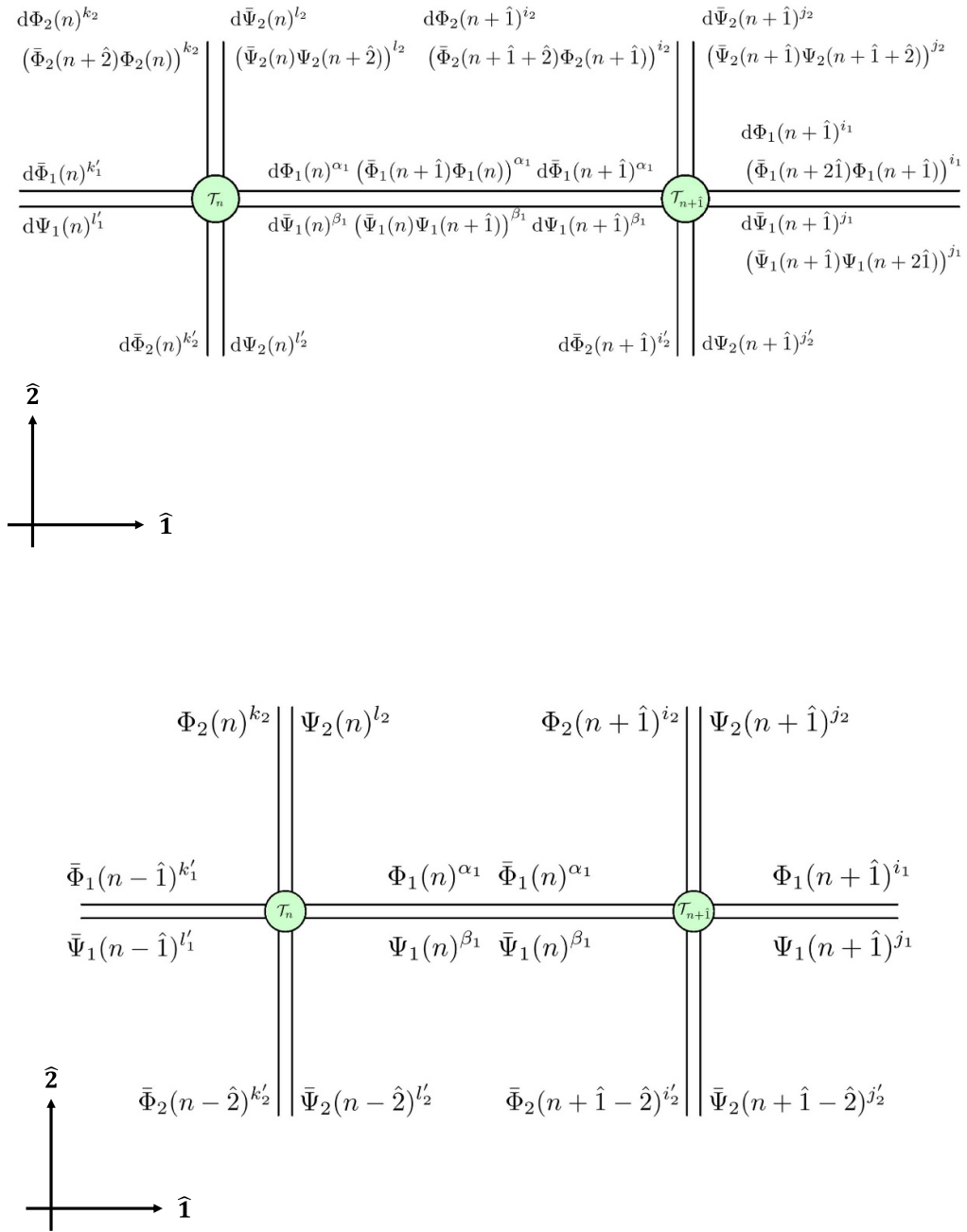


Figure 2. Graphical representation of the exact contraction between $\mathcal{T}_{n+1}$ and $\mathcal{T}_n$. (Top) Illustration of equation (20). Internal lines denote the integration over the auxiliary Grassmann variables and summation over α_1 and β_1 . (Bottom) Illustration of equation (30). Internal lines denote the weighted integrals $\int_{\bar{\Phi}_1(n), \Phi_1(n)}$ and $\int_{\bar{\Psi}_1(n), \Psi_1(n)}$ defined by equation (17). We have used the common Greek indices for the contracted auxiliary Grassmann variables because of equation (32).

explicitly showing the site dependence of auxiliary Grassmann variables.

Note that the sign factor in equation (21) can be further simplified by rearranging the Grassmann measures in equation (22). Instead of equation (23), one can find

$$\sum \int \mathcal{T}_{n+1} \mathcal{T}_n = M'_{(i_1 j_1)(i_2 j_2)(k_2 l_2)(k'_1 l'_1)(i'_2 j'_2)(k'_2 l'_2)} \times \mathcal{Q}'_{(i_1 j_1)(i_2 j_2)(k_2 l_2)(k'_1 l'_1)(k'_2 l'_2)(i'_2 j'_2)}, \quad (25)$$

where

$$\begin{aligned} M'_{(i_1 j_1)(i_2 j_2)(k_2 l_2)(k'_1 l'_1)(i'_2 j'_2)(k'_2 l'_2)} &= \sum_{\alpha_1, \beta_1} (-1)^{\alpha_1} T_{n+1; (i_1 j_1)(i_2 j_2)(\alpha_1 \beta_1)(i'_2 j'_2)} \\ &\times T_{n; (\alpha_1 \beta_1)(k_2 l_2)(k'_1 l'_1)(k'_2 l'_2)}, \quad (26) \\ \mathcal{Q}'_{(i_1 j_1)(i_2 j_2)(k_2 l_2)(k'_1 l'_1)(i'_2 j'_2)(k'_2 l'_2)} &= d\Phi_1^{i_1} d\bar{\Phi}_1^{j_1} d\Psi_2^{i_2} d\bar{\Psi}_2^{j_2} d\Psi_2^{k_2} d\bar{\Psi}_2^{l_2} d\Psi_1^{i'_2} d\bar{\Psi}_1^{j'_2} d\bar{\Phi}_1^{k'_1} d\Psi_2^{k'_2} d\bar{\Phi}_2^{l'_2} d\Psi_2^{i'_2} d\bar{\Phi}_2^{j'_2} \\ &\times (\bar{\Phi}_1 \Phi_1)^{i_1} (\bar{\Psi}_1 \Psi_1)^{j_1} (\bar{\Phi}_2 \Phi_2)^{i_2} (\bar{\Psi}_2 \Psi_2)^{j_2} (\bar{\Phi}_2 \Phi_2)^{k_2} (\bar{\Psi}_2 \Psi_2)^{l_2}. \quad (27) \end{aligned}$$

The identity used here is

$$(-1)^{(i'_2 + j'_2)(k_2 + l_2 + k'_1 + l'_1 + k'_2 + l'_2)} = (-1)^{(i'_2 + j'_2)(\alpha_1 + \beta_1)}, \quad (28)$$

because the fundamental tensor $\mathcal{T}_n$ is Grassmann-even; $T_{n; (\alpha_1 \beta_1)(k_2 l_2)(k'_1 l'_1)(k'_2 l'_2)}$ takes a non-zero value only if

$$\alpha_1 + \beta_1 + k_2 + l_2 + k'_1 + l'_1 + k'_2 + l'_2 \pmod{2} = 0 \quad (29)$$

is satisfied. This kind of identity is useful to simplify the sign factor arising in the exact contraction between the fundamental tensors.

Let us consider the same contraction but with equation (15). Since $\mathcal{T}_{n+1}$ and $\mathcal{T}_n$ are connected via $\Phi_1(n)$, $\bar{\Phi}_1(n)$, $\Psi_1(n)$, and $\bar{\Psi}_1(n)$, they should be integrated. The exact contraction gives

$$\begin{aligned} &\int \bar{\Phi}_1(n), \Phi_1(n) \int \bar{\Psi}_1(n), \Psi_1(n) \mathcal{T}_{n+1; \Phi_1 \Psi_1 \Phi_2 \Psi_2 \bar{\Psi}_1(n) \bar{\Phi}_1(n) \bar{\Psi}_2 \bar{\Phi}_2} \\ &\times \mathcal{T}_{n; \Phi_1(n) \Psi_1(n) \Phi_2 \Psi_2 \bar{\Psi}_1 \bar{\Phi}_1 \bar{\Psi}_2 \bar{\Phi}_2} \\ &= \sum M_{(i_1 j_1)(i_2 j_2)(i'_2 j'_2)(k_2 l_2)(k'_1 l'_1)(k'_2 l'_2)} \Phi_1^{i_1} \Psi_1^{j_1} \Phi_2^{i_2} \Psi_2^{j_2} \bar{\Psi}_2^{i'_2} \bar{\Phi}_2^{j'_2} \\ &\times \Phi_2^{k_2} \Psi_2^{l_2} \bar{\Psi}_1^{k'_1} \bar{\Phi}_1^{l'_1} \bar{\Psi}_2^{k'_2} \bar{\Phi}_2^{l'_2}, \quad (30) \end{aligned}$$

with

$$\begin{aligned} M_{(i_1 j_1)(i_2 j_2)(i'_2 j'_2)(k_2 l_2)(k'_1 l'_1)(k'_2 l'_2)} &= \sum_{\alpha_1, \beta_1} (-1)^{(\alpha_1 + \beta_1)(i'_2 + j'_2) + \alpha_1 + \beta_1} T_{n+1; (i_1 j_1)(i_2 j_2)(\alpha_1 \beta_1)(i'_2 j'_2)} \\ &\times T_{n; (\alpha_1 \beta_1)(k_2 l_2)(k'_1 l'_1)(k'_2 l'_2)}. \quad (31) \end{aligned}$$

The right-hand side of equation (30) defines a new Grassmann tensor. Here, the contraction between the coefficient tensors

should be understood as a result of the integrals on the auxiliary Grassmann variables. From equation (17), we find that

$$\int_{\bar{\Theta}, \Theta} \Theta^i \bar{\Theta}^j = \delta_{ij}, \quad (32)$$

where Θ and $\bar{\Theta}$ are the Grassmann variables. The integration of the auxiliary variables naturally introduces contractions between the corresponding coefficient tensors in this formalism. Figure 2 illustrates the exact contraction in the left-hand side of equation (30), without omitting the site dependence of auxiliary Grassmann variables.

Rearranging the auxiliary Grassmann variables, the right-hand side of equation (30) can be

$$\begin{aligned} &\sum M'_{(i_1 j_1)(i_2 j_2)(k_2 l_2)(k'_1 l'_1)(i'_2 j'_2)(k'_2 l'_2)} \Phi_1^{i_1} \Psi_1^{j_1} \Phi_2^{i_2} \Psi_2^{j_2} \Phi_2^{k_2} \Psi_2^{l_2} \\ &\times \bar{\Psi}_1^{l'_1} \bar{\Phi}_1^{k'_1} \bar{\Psi}_2^{l'_2} \bar{\Phi}_2^{k'_2} \bar{\Psi}_2^{j'_2} \bar{\Phi}_2^{i'_2}, \quad (33) \end{aligned}$$

where

$$\begin{aligned} M'_{(i_1 j_1)(i_2 j_2)(k_2 l_2)(k'_1 l'_1)(i'_2 j'_2)(k'_2 l'_2)} &= \sum_{\alpha_1, \beta_1} (-1)^{\alpha_1 + \beta_1} T_{n+1; (i_1 j_1)(i_2 j_2)(\alpha_1 \beta_1)(i'_2 j'_2)} \\ &\times T_{n; (\alpha_1 \beta_1)(k_2 l_2)(k'_1 l'_1)(k'_2 l'_2)}. \quad (34) \end{aligned}$$

The same identity as in equations (28) and (29) has been utilized.

Comparing equations (24) and (31), or equations (26) and (34), we see that the difference between the signs obtained in the two formulations is $(-1)^{\beta_1}$. This can be explained by the different ways to decompose the hopping terms employed in the two formulations. In equations (3) and (4), the last factors built just by auxiliary Grassmann variables inherit the structure of original hopping terms. On the other hand, there is no such difference between equations (12) and (13). The corresponding sign factor $(-1)^{j_1}$ has already been included in the coefficient tensor as in equation (16).

If we performed the exact contractions using the methods described so far, the path integral shown in equations (10) and (19) could be obtained exactly. However, in practice, it is impossible to keep carrying out the exact contractions when we consider the model in an arbitrarily large volume. This is because the number of fundamental tensors in equations (10) and (19) is the same as the number of lattice sites in Λ . Therefore, we need to consider performing the contraction not exactly but approximately. Several TRG schemes are reviewed in sections 2.5 and 2.6.

2.3. Model-independent notation

In this subsection, we introduce a new notation for the fundamental tensor

$$\mathcal{T}_n = T_{n; xtx't'} \mathcal{G}_{n; xtx't'}, \quad (35)$$

instead of equation (6). The variable x is defined by two bits via $x = (i_1 j_1)$ and $t, x',$ and t' are defined similarly. $\mathcal{G}_{n; xtx't'}$ is

defined by the right-hand side of equation (8). The expression in equation (35) may seem fine because it does not depend on the details of our model, the number of components in the original fermionic field or the structure of hopping terms, and it only depends on the lattice geometry. However, this notation can be problematic when considering their exact contractions. As we have observed in equation (24), or equation (26), the sign factor from the auxiliary Grassmann integrals does distinguish the forward and backward hopping terms in the formulation in [102]. One of the ways to resolve this issue is to modify equation (4) as

$$\begin{aligned} & e^{t\bar{\psi}(n)\psi(n+\hat{\nu})} \\ &= \sum_{j_\nu(n)=0}^1 (\sqrt{t}\bar{\psi}(n)d\bar{\Psi}_\nu(n))^{j_\nu(n)} (\sqrt{t}\psi(n+\hat{\nu})d\Psi_\nu(n+\hat{\nu}))^{j_\nu(n)} \\ & \quad \times (-\Psi_\nu(n+\hat{\nu})\bar{\Psi}_\nu(n))^{j_\nu(n)}, \end{aligned} \quad (36)$$

and absorbing the extra factor $(-1)^{j_\nu(n)}$ into the bosonic tensor in equation (7). Note that this modification makes the bosonic tensor in equation (7) exactly the same as the coefficient tensor in equation (16). From now on, we identify the expression in equation (35) with this modification. The exact contraction demonstrated in section 2.2 is then denoted by

$$\sum \int \mathcal{T}_{n+\hat{1}} \mathcal{T}_n = \sum_{\alpha} T_{n+\hat{1};x\alpha t_1'} T_{n;\alpha t_2 x' t_2'} \int \mathcal{G}_{n+\hat{1};x\alpha t_1'} \mathcal{G}_{n;\alpha t_2 x' t_2'}. \quad (37)$$

Now, we introduce the Grassmann parity function f_x for $x = (i_1 j_1)$ such that

$$f_x = i_1 + j_1 \pmod{2}, \quad (38)$$

and $f_t, f_{x'},$ and $f_{t'}$ are done in the same way. Using these parity functions, equation (37) is evaluated as

$$\sum \int \mathcal{T}_{n+\hat{1}} \mathcal{T}_n = M_{x t_1 t_2 x' t_1' t_2'} \mathcal{Q}_{x t_1 t_2 x' t_1' t_2'}, \quad (39)$$

where

$$M_{x t_1 t_2 x' t_1' t_2'} = \sum_{\alpha} (-1)^{f_{\alpha}} T_{n+\hat{1};x\alpha t_1'} T_{n;\alpha t_2 x' t_2'}, \quad (40)$$

and

$$\begin{aligned} & \mathcal{Q}_{x t_1 t_2 x' t_1' t_2'} \\ &= d\Phi_1^{i_1} d\bar{\Phi}_1^{j_1} d\Psi_2^{i_2} d\bar{\Psi}_2^{j_2} d\Psi_1^{k_1} d\bar{\Psi}_1^{j_1'} d\bar{\Phi}_1^{k_1'} d\Psi_2^{i_2'} d\bar{\Phi}_2^{k_2'} d\Psi_2^{j_2'} d\bar{\Phi}_2^{i_2'} \\ & \quad \times (\bar{\Phi}_1 \Phi_1)^{i_1} (\Psi_1 \bar{\Psi}_1)^{j_1} (\bar{\Phi}_2 \Phi_2)^{i_2} (\Psi_2 \bar{\Psi}_2)^{j_2} (\bar{\Phi}_2 \Phi_2)^{k_2} (\Psi_2 \bar{\Psi}_2)^{i_2'}. \end{aligned} \quad (41)$$

It should be emphasized that the exact contraction between the fundamental tensors expressed as in equation (35) is straightforwardly generalized to other models just modifying the definition of the Grassmann parity function in equation (38).

Similarly, we introduce the following expression for equation (15),

$$\mathcal{T}_{n;XT\bar{X}\bar{T}} = \sum_{x,t,x',t'} T_{n;xtx't'} X^x T^t \bar{X}^{x'} \bar{T}^{t'}. \quad (42)$$

As in equation (35), x is defined by two bits via $x = (i_1 j_1)$ and $t, x',$ and t' are defined similarly. X can be regarded as a two-component auxiliary Grassmann variables via $X = (\Phi_1, \Psi_1)$ and X^x as $X^x = \Phi_1^{i_1} \Psi_1^{j_1}$. $\bar{X}$ can also be regarded like $\bar{X} = (\bar{\Phi}_1, \bar{\Psi}_1)$, but $\bar{X}^x$ should be understood as $\bar{X}^x = \bar{\Psi}_1^{i_1'} \bar{\Phi}_1^{j_1'}$. T ($\bar{T}$) and T^t ($\bar{T}^{t'}$) are defined in the same way. Equation (30) can be equivalently expressed as

$$\begin{aligned} & \int_{\Theta, \Theta} \mathcal{T}_{n+\hat{1};xT_1\Theta\bar{T}_1} \mathcal{T}_{n;\Theta T_2\bar{X}\bar{T}_2} \\ &= \sum_{x,t_1,t_2,x',t_1',t_2'} M_{x t_1 t_2 x' t_1' t_2'} X^x T_1^{t_1'} T_2^{t_2'} \bar{X}^{x'} \bar{T}_2^{t_2'} \bar{T}_1^{t_1'}, \end{aligned} \quad (43)$$

where the coefficient tensor $M_{x t_1 t_2 x' t_1' t_2'}$ is defined in the exactly same way with equation (40) using the Grassmann parity function defined in equation (38). Note that Θ and $\bar{\Theta}$ in equation (43) are two-component Grassmann variables and $\int_{\Theta, \Theta}$ in the left-hand side is defined by

$$\int_{\Theta, \Theta} = \prod_i \int d\bar{\Theta}_i d\Theta_i e^{-\bar{\Theta}_i \Theta_i}, \quad (44)$$

which is a natural extension of equation (17). We can also see that f_x in equation (38) counts the Grassmann parity of X^x . The right-hand side of equation (43) is the Grassmann tensor that can be written as $\mathcal{M}_{x t_1 t_2 x' t_1' t_2'}$ in our notation. Extend the definition of the Grassmann parity function in equation (38), and the notation in equation (42) immediately allows us to deal with the case where the original fermionic model is described by the multi-component Grassmann variables.

So far, we have confirmed that the bosonic tensor derived by [102] and the coefficient tensor by [125] can be equivalent if we slightly modify the formulation of [102]. It has also been confirmed that the exact contraction can be described similarly using the same Grassmann parity function with either formulation. Therefore, these two formulations are fully equivalent in terms of memory usage and computational time for contractions. Figure 3 graphically shows the Grassmann tensor network representation of equation (2). Figure 3(A) can be interpreted in two ways following the notation rule in figures 1 and 2. Figure 3(B) assumes the model-independent notation. The following demonstration always assumes the formalism in [125] with the notation introduced in this section 2.3.

2.4. Extension to lattice gauge theories

Current Grassmann tensor network formulations are easily combined with the usual tensor network formulations for pure gauge or bosonic theories. See [67] and several reviews [74, 130–132] for more details on how to construct fundamental

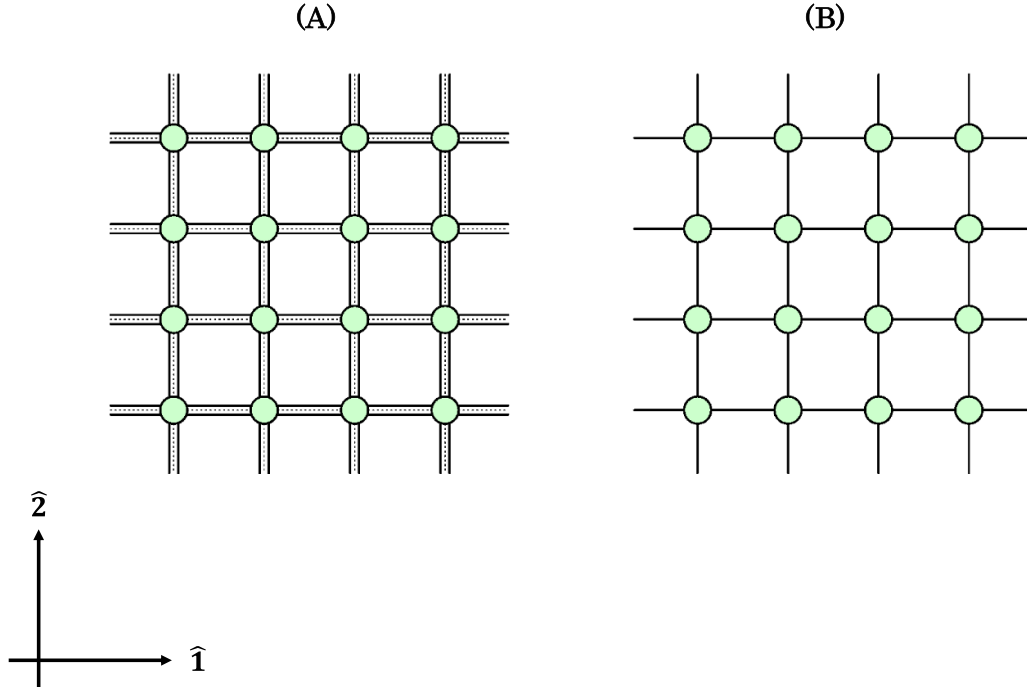


Figure 3. Diagrammatic representation of Grassmann tensor network on a two-dimensional square lattice. (A) Grassmann tensor network representation of equation (2). Interpretation of each line is given in figure 1. Background dotted lines denote the real-space lattice. (B) Model-independent description of two-dimensional Grassmann tensor network. The shape of each fundamental tensor is determined only by the lattice geometry.

tensors in these theories. Here, let us consider the two-dimensional $\mathbb{Z}_N$ gauge theory on a periodic square lattice defined by

$$S = S_g + S_f, \quad (45)$$

as an example. We assume the standard Wilson action,

$$S_g = -\beta \sum_{n \in \Lambda} \Re \left[U_1(n) U_2(n + \hat{1}) U_1^*(n + \hat{2}) U_2^*(n) \right], \quad (46)$$

and the staggered fermion action,

$$S_f = \sum_{n, \nu} \frac{\eta_\nu(n)}{2} \left[\bar{\chi}(n) U_\nu(n) \chi(n + \hat{\nu}) - \bar{\chi}(n + \hat{\nu}) U_\nu^*(n) \chi(n) \right] + m \sum_n \bar{\chi}(n) \chi(n). \quad (47)$$

β and m denote the inverse gauge coupling and mass, respectively. The staggered sign function $\eta_\nu(n)$ is defined by $\eta_1(n) = 1$ and $\eta_2(n) = (-1)^{n_1}$ at each site $n = (n_1, n_2)$. Our goal is to derive the Grassmann tensor network representation for the following path integral,

$$Z = \int \prod_{n, \nu} dU_\nu(n) \prod_n d\chi(n) d\bar{\chi}(n) e^{-S}. \quad (48)$$

We parameterize $U_\nu(n)$ by an integer $q_\nu(n)$ mod N via $U_\nu(n) = \exp[2\pi i q_\nu(n)/N]$. Following section 2.1, one can immediately obtain a Grassmann tensor with some integer indices. We begin with decomposing the hopping terms via

$$\begin{aligned} & e^{\eta_\nu(n) \bar{\chi}(n + \hat{\nu}) U_\nu^*(n) \chi(n) / 2} \\ &= \int d\bar{\Phi}_\nu(n) d\Phi_\nu(n) e^{-\bar{\Phi}_\nu(n) \Phi_\nu(n)} \\ & \quad \times e^{\bar{\chi}(n + \hat{\nu}) \bar{\Phi}_\nu(n) / \sqrt{2}} e^{\eta_\nu(n) U_\nu^*(n) \chi(n) \Phi_\nu(n) / \sqrt{2}}, \end{aligned} \quad (49)$$

$$\begin{aligned} & e^{-\eta_\nu(n) \bar{\chi}(n) U_\nu(n) \chi(n + \hat{\nu}) / 2} \\ &= \int d\bar{\Psi}_\nu(n) d\Psi_\nu(n) e^{-\bar{\Psi}_\nu(n) \Psi_\nu(n)} \\ & \quad \times e^{\eta_\nu(n) U_\nu(n) \bar{\chi}(n) \Psi_\nu(n) / \sqrt{2}} e^{\chi(n + \hat{\nu}) \bar{\Psi}_\nu(n) / \sqrt{2}}. \end{aligned} \quad (50)$$

Integrating out the original staggered fields at each site, one obtains

$$\begin{aligned} & \int d\chi d\bar{\chi} e^{-m \bar{\chi} \chi} \prod_\nu e^{\bar{\chi} \bar{\Phi}_\nu(n - \hat{\nu}) / \sqrt{2}} e^{\eta_\nu(n) U_\nu(n) \bar{\chi} \Psi_\nu(n) / \sqrt{2}} \\ & \quad \times e^{\eta_\nu(n) U_\nu^*(n) \chi \Phi_\nu(n) / \sqrt{2}} e^{\chi \bar{\Psi}_\nu(n - \hat{\nu}) / \sqrt{2}} \\ &= \sum_{x_f, t_f, x'_f, t'_f} T_{x_f t_f, x'_f t'_f, q_2(n) q_1(n)}^{(f)} X^{x_f} T^{t_f} \bar{X}^{x'_f} \bar{T}^{t'_f}, \end{aligned} \quad (51)$$

where we have employed the model-independent notation in section 2.3 and the coefficient tensor is given by

$$\begin{aligned}
 T_{x_f t_f x'_f t'_f, q_2(n) q_1(n)}^{(f)} &= T_{(i_1 j_1)(i_2 j_2)(i'_1 j'_1)(i'_2 j'_2), q_2(n) q_1(n)}^{(f)} \\
 &= \sqrt{2}^{-\sum_\nu (i_\nu + j_\nu + i'_\nu + j'_\nu)} (-1)^{j_1(i_2 + j'_1 + j'_2) + j_2(j'_1 + j'_2) + i'_1 j'_2 + n_1(i_2 + j_2)} \\
 &\quad \times e^{\frac{2\pi i}{N} \sum_\nu q_\nu(n)(j_\nu - i_\nu)} \\
 &\quad \times \left[-m \delta_{i_1 + i_2 + j'_1 + j'_2, 0} \delta_{i'_1 + i'_2 + j_1 + j_2, 0} \right. \\
 &\quad \left. + \delta_{i_1 + i_2 + j'_1 + j'_2, 1} \delta_{i'_1 + i'_2 + j_1 + j_2, 1} \right]. \quad (52)
 \end{aligned}$$

Note that X^{x_f} , T^{t_f} , $\bar{X}^{x'_f}$, and $\bar{T}^{t'_f}$ in the right-hand side of equation (51) have been defined in the same way with equation (42). We also introduce a four-leg tensor to describe the plaquette interaction term in $e^{-\delta_g}$ via

$$\begin{aligned}
 T_{q_2(n+\hat{1}) q_1(n+\hat{2}) q_2(n) q_1(n)}^{(g)} &= \exp \left[\beta \cos \left\{ \frac{2\pi}{N} \left(q_1(n) + q_2(n+\hat{1}) \right. \right. \right. \\
 &\quad \left. \left. \left. - q_1(n+\hat{2}) - q_2(n) \right) \right\} \right], \quad (53)
 \end{aligned}$$

following [133]. Regarding q_ν 's as tensor subscripts, we now define the fundamental tensor associated at the lattice site n as

$$\mathcal{T}_{n; XT \bar{X} \bar{T}, x_g t_g x'_g t'_g} = \sum_{x_f, t_f, x'_f, t'_f} T_{n; x_f t_f x'_f t'_f, x_g t_g x'_g t'_g} X^{x_f} T^{t_f} \bar{X}^{x'_f} \bar{T}^{t'_f}, \quad (54)$$

where

$$T_{n; x_f t_f x'_f t'_f, x_g t_g x'_g t'_g} = T_{x_f t_f x'_f t'_f, x_g t_g}^{(f)} \cdot T_{x_g t_g x'_g t'_g}^{(g)}. \quad (55)$$

The path integral in equation (48) is now represented by

$$Z = \text{gTr} \left[\prod_n \mathcal{T}_n \right]. \quad (56)$$

Here, 'gTr' stands not only for the integrations over all auxiliary Grassmann variables but also for the summations over all integers corresponding to the link variables. Analogously, we refer $\mathcal{T}_n$ and T_n in equation (54) to the Grassmann tensor and its coefficient tensor, respectively. See figure 4 for the diagrammatic explanation.

Let us briefly see how the exact contraction is carried out between these fundamental tensors. As in section 2.2, we consider the contraction between the fundamental tensors at $n + \hat{1}$ and n for demonstration. Since a link variable is shared between these two fundamental tensors, the exact contraction should be

$$\sum_{\alpha_g} \int_{\bar{\Theta}, \Theta} \mathcal{T}_{n+\hat{1}; XT \bar{X} \bar{T}, x_g t_g x'_g t'_g} \mathcal{T}_{n; \Theta T_2 \bar{X} \bar{T}_2, \alpha_g t_{g2} x'_{g2} t'_{g2}}. \quad (57)$$

One will immediately find that the above contraction results in a new Grassmann tensor such that,

$$\begin{aligned}
 \mathcal{M}_{XT_1 T_2 \bar{X} \bar{T}_2 \bar{T}_1, x_g t_{g1} t'_{g2} x'_{g1} t'_{g2}} &= \sum_{x_f, t_f, x'_f, t'_f} M_{x_f t_f x'_f t'_f, t'_{f2}, x_g t_{g1} t'_{g2} x'_{g1} t'_{g2}} \\
 &\quad \times X^{x_f} T^{t_f} T^{t'_{f2}} \bar{X}^{x'_f} \bar{T}^{t'_{f2}} \bar{T}^{t'_{f1}}, \quad (58)
 \end{aligned}$$

whose coefficient tensor is defined by

$$\begin{aligned}
 M_{x_f t_f x'_f t'_f, t'_{f2}, x_g t_{g1} t'_{g2} x'_{g1} t'_{g2}} &= \sum_{\alpha_f, \alpha_g} (-1)^{f_{\alpha_f}} T_{n+\hat{1}; x_f t_f x'_f t'_f, x_g t_{g1} \alpha_g t'_{g1}} \\
 &\quad \times T_{n; \alpha_f t_{f2} x'_f t'_{f2}, \alpha_g t_{g2} x'_{g2} t'_{g2}}. \quad (59)
 \end{aligned}$$

The Grassmann parity function f_{α_f} is given in the same way with equation (38). Introducing a super index $p = (p_f, p_g)$ for $p = x, t, x', t'$, equation (59) reads

$$M_{x t_1 t_2 x' t'_1 t'_2} = \sum_{\alpha} (-1)^{F_{\alpha}} T_{n+\hat{1}; x t_1 \alpha t'_1} T_{n; \alpha t_2 x' t'_2}, \quad (60)$$

where we have defined a new parity function for the super index $\alpha = (\alpha_f, \alpha_g)$ by

$$F_{\alpha} = f_{\alpha_f}. \quad (61)$$

The function F_{α} tells us the Grassmann parity of the super index α : when $F_{\alpha} = 0$ (1), the super index α is describing the Grassmann-even (odd) contribution.

Therefore, we reach the important conclusion that the structure of the Grassmann tensor network and the contraction rule are not affected by the gauge fields. The resulting coefficient tensor in equation (60) has the same expression as equation (40), which was the coefficient tensor for the pure fermionic model in equation (1). The exact contractions among the Grassmann tensors result in those among the coefficient tensors, with or without the lattice gauge fields. One can use the tensor network diagram in figure 3(B) instead of figure 4(B) to represent equation (56). Although we have assumed $\mathbb{Z}_N$ as a gauge group for simplicity, the above construction can be combined with other tensor network formulations for various lattice gauge theories including the non-Abelian fields [67, 134–139].

2.5. Approximate contraction by the Levin–Nave TRG

The usual TRG algorithms are intended to be applied to the classical systems without any Grassmann variables. Still, any TRG algorithm can be utilized to evaluate the Grassmann path integrals. As we have seen, the contractions between the Grassmann tensors always result in those between the corresponding coefficient tensors with some sign factors arising from the integrals over the auxiliary Grassmann variables. Therefore, we can use the TRG algorithms to carry out the contractions among the coefficient tensors approximately. In this subsection, we demonstrate how to extend the Levin–Nave TRG [27] for the Grassmann tensor networks. The case of the HOTRG [32] is discussed in the next subsection.

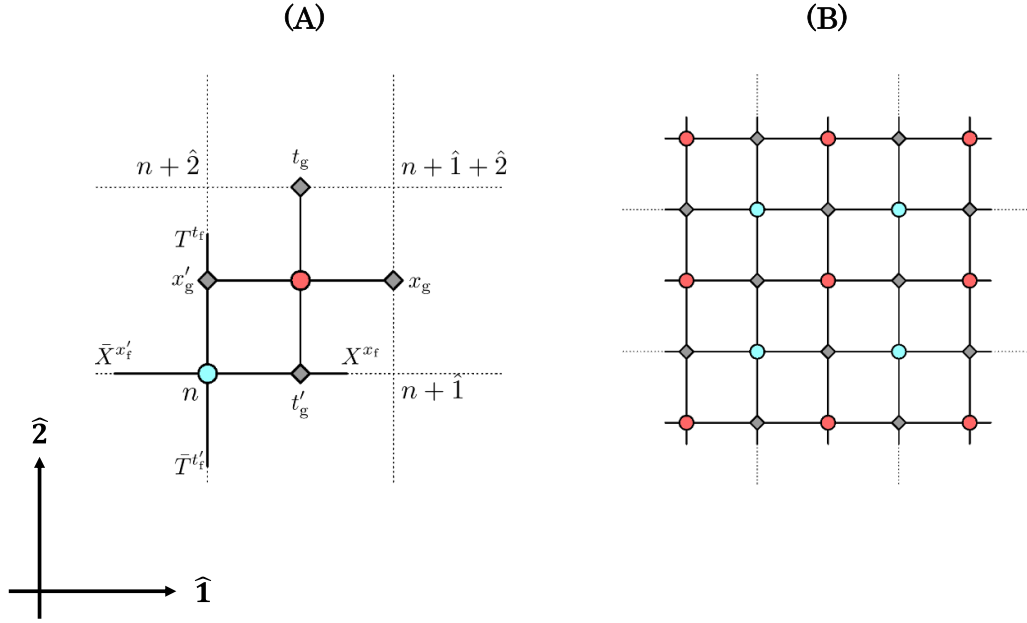


Figure 4. Diagrammatic representation of Grassmann tensor network formulation for two-dimensional lattice gauge theories. Background dotted lines show a real-space square lattice. (A) Structure of the fundamental tensor defined in equation (54). The blue symbol is located on the site and denotes the right-hand side of equation (51). Note that the coefficient tensor depends on n_1 as shown in equation (52). The red symbol is located on the plaquette and denotes the four-leg tensor $T_{x_g t_g x'_g t'_g}^{(g)}$ in equation (53). Each diamond shows the link variable. (B) Grassmann tensor network in equation (56). Each internal line represents the contraction between auxiliary Grassmann variables and integration over the shared link variable.

Let us begin with reviewing the original Levin–Nave TRG. The algorithm aims to evaluate the partition function or path integral represented by the tensor network

$$Z = \text{tTr} \left[\prod_{n \in \Lambda} T_n \right], \quad (62)$$

where we assume the two-dimensional periodic square lattice Λ and T_n is a four-leg tensor defined on the site n . The algorithm employs the singular value decomposition (SVD) to decimate the four-leg tensor T_n into three-leg tensors:

$$T_{n;xtx't'} \simeq \sum_{a=1}^{D_{\text{LNTRG}}} A_{n;xta} B_{n;ax't'}, \quad (63)$$

$$T_{n;xtx't'} \simeq \sum_{a=1}^{D_{\text{LNTRG}}} C_{n;xt'a} D_{n;ax't'}. \quad (64)$$

Each three-leg tensor is defined as a unitary matrix multiplied by the square root of its singular value. In the above expressions, we have assumed that the singular values are in descending order. The truncation parameter D_{LNTRG} is called the bond dimension. Since we have used the SVD, equations (63) and (64) give the best approximation in terms of the Frobenius norm of T_n under the fixed bond dimension. Then, we define a new four-leg tensor via

$$T_{n';xtx't'} = \sum_{x_1, x_2, t_1, t_2} C_{n+2; x_2 t_1 x} A_{n; x_1 t_1 t} D_{n+1; x' x_1 t_2} B_{n+1+2; t' x_2 t_2}, \quad (65)$$

which approximates the original Z as

$$Z \simeq \text{tTr} \left[\prod_{n' \in \Lambda'} T_{n'} \right]. \quad (66)$$

Therefore, $T_{n'}$ is a new fundamental tensor defined on a site n' in the coarse-grained lattice Λ' . Since equation (66) describes the tensor network whose geometry is the same as that in equation (62), we can easily repeat the above decimation procedure. Repeating this procedure N times, 2^N original fundamental tensors are approximately contracted. Thanks to this property, the algorithm allows us to evaluate the partition function in the thermodynamic limit. The Levin–Nave TRG requires the $O(D_{\text{LNTRG}}^6)$ complexity and the $O(D_{\text{LNTRG}}^4)$ memory cost. Note that the cost can further be reduced to the $O(D_{\text{LNTRG}}^5)$ complexity and the $O(D_{\text{LNTRG}}^3)$ memory using the randomized SVD [140]. The algorithm is graphically summarized in figure 5.

We now need to define the SVD for the Grassmann tensor to extend the algorithm for the Grassmann tensor network. Suppose $\mathcal{O}_{\Phi\Psi}$ is a Grassmann tensor defined via

$$\mathcal{O}_{\Phi\Psi} = \sum_{i,j} \mathcal{O}_{ij} \Phi^i \Psi^j, \quad (67)$$

where we have assumed the notation in section 2.3, say $i = (i_1 \cdots i_m)$ and $j = (j_1 \cdots j_n)$ with $\Phi^i = \Phi_1^{i_1} \cdots \Phi_m^{i_m}$ and $\Psi^j = \Psi_1^{j_1} \cdots \Psi_n^{j_n}$. We also assume that $\mathcal{O}$ is Grassmann-even, in other words, the matrix elements are zero unless the sum of all the

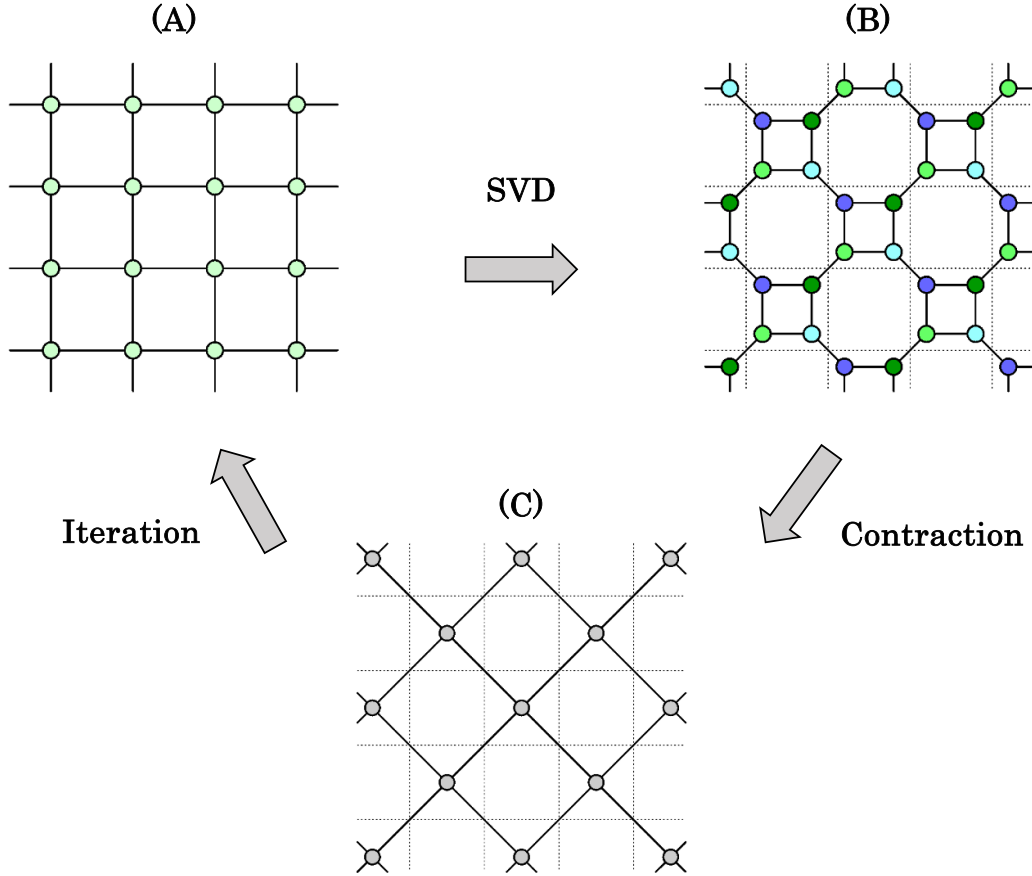


Figure 5. Schematic illustration of the Levin–Nave TRG algorithm. Background dotted lines show a real-space square lattice. (A) Initial tensor network on the lattice. (B) Two kinds of SVD as shown in equations (63) and (64). (C) New tensor network by contracting four three-leg tensors.

indices (i and j) is even. Since the coefficient tensor is normal, we can apply the SVD for O_{ij} ,

$$O_{ij} = \sum_{a=1}^{\min(2^m, 2^n)} U_{ia} \sigma_a V_{aj}^\dagger = \sum_{a,b} A_{ia} \delta_{ab} B_{bj}. \quad (68)$$

In the second equality, we set $A = U\sqrt{\sigma}$ and $B = \sqrt{\sigma}V^\dagger$. Recalling the identity in equation (32), we can express δ_{ab} by introducing new auxiliary Grassmann variables. Plugging equation (68) into equation (67), we now have the SVD for the Grassmann tensor $\mathcal{O}$ as

$$\mathcal{O}_{\Phi\Psi} = \int_{\Xi, \bar{\Xi}} \mathcal{A}_{\Phi\Xi} \mathcal{B}_{\bar{\Xi}\Psi}, \quad (69)$$

where Ξ and $\bar{\Xi}$ are the $\min(m, n)$ -component auxiliary Grassmann variables introduced via equation (32). The Grassmann tensors $\mathcal{A}$ and $\mathcal{B}$ in the right-hand side have A and B in equation (68) as their coefficient tensors. Since we have assumed that $\mathcal{O}$ is Grassmann-even, the resulting $\mathcal{A}$ and $\mathcal{B}$ are also Grassmann-even, as we will see in section 2.7. It is no exaggeration to say that the SVD for a Grassmann tensor is the SVD for its coefficient tensor. There is no difficulty in

introducing the low-rank approximation based on the SVD. In the following, we use the approximation

$$\mathcal{O}_{\Phi\Psi} \simeq \int_{\Xi, \bar{\Xi}}^{D_{\text{LNTRG}}} \mathcal{A}_{\Phi\Xi} \mathcal{B}_{\bar{\Xi}\Psi}, \quad (70)$$

which means that the SVD of the coefficient tensor is truncated up to the bond dimension D_{LNTRG} as

$$O_{ij} \simeq \sum_{a,b=1}^{D_{\text{LNTRG}}} A_{ia} \delta_{ab} B_{bj}. \quad (71)$$

We will make some practical remarks on the SVD of the Grassmann tensor in section 2.7

We now extend the algorithm for evaluating

$$Z = \text{gTr} \left[\prod_{n \in \Lambda} \mathcal{T}_n \right], \quad (72)$$

with the Grassmann tensor $\mathcal{T}_{n;XT\bar{X}\bar{T}}$. We can formally write the SVD for the Grassmann tensor corresponding to equations (63) and (64) as

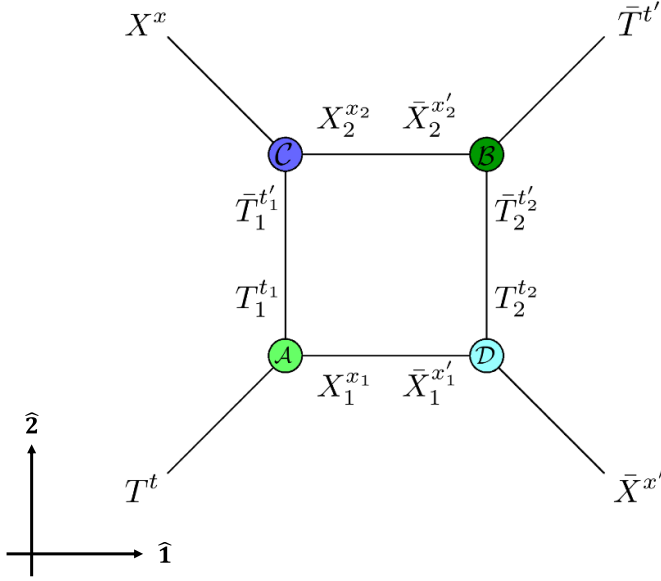


Figure 6. Diagrammatic representation of equation (75).

$$\mathcal{T}_{n;XT\bar{X}\bar{T}} \simeq \int_{\Xi, \Xi}^{D_{\text{LNTRG}}} \mathcal{A}_{n;XT\Xi} \mathcal{B}_{n;\Xi\bar{X}\bar{T}}, \quad (73)$$

$$\mathcal{T}_{n;XT\bar{X}\bar{T}} \simeq \int_{\Xi, \Xi}^{D_{\text{LNTRG}}} \mathcal{C}_{n;XT\Xi} \mathcal{D}_{n;\Xi\bar{X}\bar{T}}. \quad (74)$$

When $\mathcal{T}_n$ is Grassmann-even, then $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$, and $\mathcal{D}$ are also Grassmann-even. Therefore, we can easily define a new Grassmann tensor from the contractions among them,

$$\begin{aligned} \mathcal{T}_{n';XT\bar{X}\bar{T}} &= \int_{\bar{X}_1, X_1} \int_{\bar{X}_2, X_2} \int_{\bar{T}_1, T_1} \int_{\bar{T}_2, T_2} \mathcal{C}_{n+2;X_2\bar{T}_1X} \mathcal{A}_{n;X_1T_1T} \\ &\times \mathcal{D}_{n+1;\bar{X}\bar{X}_1T_2} \mathcal{B}_{n+1+2;\bar{T}\bar{X}_2\bar{T}_2}. \end{aligned} \quad (75)$$

The above equation is analogous to equation (65) in the usual Levin–Nave TRG. Equation (75) is diagrammatically shown in figure 6. The new Grassmann tensor approximates the path integral by

$$Z \simeq \text{gTr} \left[\prod_{n' \in \Lambda'} \mathcal{T}_{n'} \right]. \quad (76)$$

The resulting Grassmann tensor network in equation (76) is identical to the previous one in equation (72), including the ordering of the Grassmann measures in gTr. Therefore, one can easily repeat the decimation procedure toward the thermodynamic limit as in the usual Levin–Nave TRG. We can understand that the SVD of the Grassmann tensor introduces new auxiliary Grassmann variables on the coarse-grained lattice Λ' . Therefore, the algorithm for the Grassmann tensor network perfectly corresponds to that for the normal tensor network as shown in figure 5. In terms of coefficient tensors, several sign factors should be included in equations (74) and (75) as we will see in section 2.7.

2.6. Approximate contraction by the HOTRG

Let us next focus on the HOTRG algorithm that applies to any d -dimensional system. The computational complexity scales with $O(D_{\text{HOTRG}}^{4d-1})$ and the memory cost does with $O(D_{\text{HOTRG}}^{2d})$, where D_{HOTRG} is the bond dimension in the HOTRG. For simplicity, we consider the two-dimensional tensor network on a periodic square lattice again. The original algorithm aims to evaluate equation (62) not decimating each local fundamental tensor but inserting the projectors that describe the coarse-graining transformation in the tensor-network language. Unlike the Levin–Nave TRG, the HOTRG performs the contraction between two adjacent fundamental tensors along each direction sequentially.

We begin with reviewing the normal HOTRG. As an example, we consider the contraction between $T_{n+\hat{1}}$ and T_n . Introducing

$$M_{xt_1t_2x't_1't_2'} = \sum_{\alpha} T_{n+\hat{1};xt_1\alpha t_1'} T_{n;\alpha t_2x't_2'}, \quad (77)$$

the HOTRG provides the coarse-graining transformation such that

$$T_{n';xt_1t_2x't_1't_2'} = \sum_{t_1, t_2, t_1', t_2'} P_{t_1t_2t} M_{xt_1t_2x't_1't_2'} Q_{t't_1't_2'}, \quad (78)$$

where the three-leg tensors P and Q are projectors to decimate the degrees of freedom. They play roles to map the original degrees of freedom $(t_1^{(')}t_2^{(')})$ to the coarse-grained one $t^{(')}$, whose size is restricted by D_{HOTRG} , the bond dimension in the algorithm. We can also regard that the HOTRG is inserting the following four-leg tensor,

$$W_{t_1t_2t_1't_2'} = \sum_{t=1}^{D_{\text{HOTRG}}} P_{t_1t_2t} Q_{tt_1't_2'}, \quad (79)$$

into the tensor network. With sufficiently large D_{HOTRG} , $W_{t_1t_2t_1't_2'}$ should be equivalent to $\delta_{t_1,t_1'}\delta_{t_2,t_2'}$ and the algorithm gives the exact contraction. $T_{n'}$ generates the approximated tensor network representation of the partition function. The coarse-graining transformation in the HOTRG can be easily repeated. As in the case of the Levin–Nave TRG, the HOTRG allows us to contract 2^N original fundamental tensors approximately, just in N times of iteration. Figure 7 graphically demonstrates the procedure of the algorithm.

There are several ways to determine P and Q . Here, we just follow the original proposal in [32] assuming the translational symmetry for the tensor network in equation (62). We consider the following two reduced density matrices,

$$\rho_{(t_1t_2)(\bar{t}_1\bar{t}_2)} = \sum_{x,x',t_1',t_2'} M_{xt_1t_2x't_1't_2'} M_{x\bar{t}_1\bar{t}_2x'\bar{t}_1'\bar{t}_2'}^*, \quad (80)$$

$$\rho_{(t_1't_2')(\bar{t}_1'\bar{t}_2')} = \sum_{x,t_1,t_2,x'} M_{xt_1t_2x't_1't_2'} M_{x\bar{t}_1\bar{t}_2x'\bar{t}_1'\bar{t}_2'}^*. \quad (81)$$

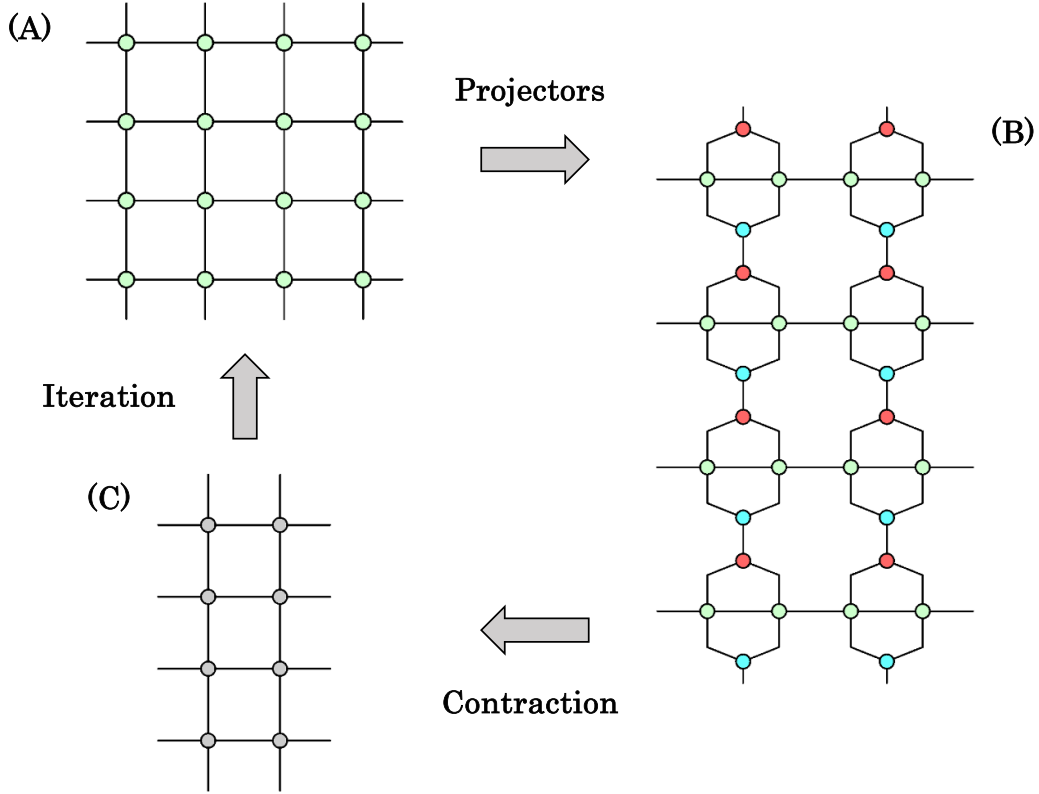


Figure 7. Schematic illustration of the HOTRG algorithm. (A) Initial tensor network on a square lattice. (B) Inset projectors into the network. Red and blue symbols show P and Q in equation (78), respectively. (C) New tensor network by contracting adjacent two fundamental tensors with two projectors.

These matrices can be decomposed as

$$\rho_{(t_1 t_2)(\bar{t}_1 \bar{t}_2)} = \sum_i U_{(t_1 t_2)i} \lambda_i U_{i(\bar{t}_1 \bar{t}_2)}^\dagger, \quad (82)$$

$$\rho'_{(t'_1 t'_2)(\bar{t}'_1 \bar{t}'_2)} = \sum_i V_{(t'_1 t'_2)i} \lambda'_i V_{i(\bar{t}'_1 \bar{t}'_2)}^\dagger, \quad (83)$$

where U and V are unitary matrices and λ and λ' denote the singular-value matrices with descending order. Defining the following quantities,

$$\epsilon^{(')} = \sum_{i > D_{\text{HOTRG}}} \lambda_i^{(')}, \quad (84)$$

we choose $P_{t_1 t_2 t} = U_{(t_1 t_2)t}^*$ and $Q_{t' t'_1 t'_2} = U_{(t'_1 t'_2)t'}$ if $\epsilon < \epsilon'$ and choose $P_{t_1 t_2 t} = V_{(t_1 t_2)t}$ and $Q_{t' t'_1 t'_2} = V_{(t'_1 t'_2)t'}$ if $\epsilon > \epsilon'$. This procedure is equivalent to the higher-order SVD (HOSVD), which is a tensorial extension of the SVD as explained in [32]. See references [141–143] for other ways to derive the optimal P and Q without assuming the translational invariance on the tensor network.

The HOTRG algorithm is straightforwardly extended to evaluate the Grassmann tensor network in equation (72). Equation (77) is correspondingly denoted by

$$\mathcal{M}_{XT_1 T_2 \bar{X} \bar{T}_1 \bar{T}_2} = \int_{\bar{\Theta}, \Theta} \mathcal{T}_{n+\hat{1}; XT_1 \bar{\Theta} \bar{T}_1} \mathcal{T}_{n; \Theta T_2 \bar{X} \bar{T}_2}, \quad (85)$$

where the coefficient tensor of $\mathcal{M}$ has been already derived in equation (40) or equation (60). The HOTRG for the Grassmann tensor network should provide us with the coarse-graining transformation such as

$$\mathcal{T}_{n'; XT_1 T_2 \bar{X} \bar{T}_1 \bar{T}_2} = \int_{\bar{T}_1, T_1} \int_{\bar{T}_2, T_2} \int_{\bar{T}'_1, T'_1} \int_{\bar{T}'_2, T'_2} \mathcal{P}_{\bar{T}_2 \bar{T}_1 T} \mathcal{M}_{XT_1 T_2 \bar{X} \bar{T}_2 \bar{T}_1 \bar{T}_2} \mathcal{Q}_{\bar{T}'_1 T'_1 T'_2}, \quad (86)$$

where $\mathcal{P}$ and $\mathcal{Q}$ are the Grassmann projectors defined by

$$\mathcal{P}_{\bar{T}_2 \bar{T}_1 T} = \sum_{t_1, t_2, t} P_{t_1 t_2 t} \bar{T}_2^{t_2} \bar{T}_1^{t_1} T^t, \quad (87)$$

$$\mathcal{Q}_{\bar{T}'_1 T'_1 T'_2} = \sum_{t'_1, t'_2, t'} Q_{t'_1 t'_2 t'} \bar{T}'_1^{t'_1} T'_1^{t'_1} T'_2^{t'_2}. \quad (88)$$

In analogy with equation (79), we can identify that the algorithm inserts

$$\mathcal{W}_{\bar{T}_2 \bar{T}_1 T'_1 T'_2} = \int_{\bar{T}, T}^{D_{\text{HOTRG}}} \mathcal{P}_{\bar{T}_2 \bar{T}_1 T} \mathcal{Q}_{\bar{T}'_1 T'_1 T'_2}, \quad (89)$$

into the Grassmann tensor network. $\mathcal{P}$ and $\mathcal{Q}$, or P and Q in other words, are determined via the same procedure in the normal HOTRG. Firstly, we define the conjugation of the Grassmann tensor by

$$(\mathcal{O}_{\Phi\Psi})^\dagger = \mathcal{O}_{\bar{\Psi}\bar{\Phi}}^* = \sum_{i,j} \mathcal{O}_{ij}^* \bar{\Psi}^j \bar{\Phi}^i, \quad (90)$$

where we have assumed that $\mathcal{O}_{\Phi\Psi}$ is defined by equation (67). Note that for $\Phi^i = \Phi_1^{i_1} \cdots \Phi_m^{i_m}$, we define $\bar{\Phi}^i$ in equation (90) as $\bar{\Phi}^i = \bar{\Phi}_m^{i_m} \cdots \bar{\Phi}_1^{i_1}$. $\bar{\Psi}^j$ is defined in the same way. Then, we define the Grassmann reduced density matrices as

$$\varrho_{T_1 T_2 \bar{T}_2 \bar{T}_1} = \int_{\bar{X}, X} \int_{X', \bar{X}'} \int_{T_2', \bar{T}_2'} \int_{T_1', \bar{T}_1'} \mathcal{M}_{X T_1 T_2 \bar{X}' \bar{T}_2' \bar{T}_1'} \mathcal{M}_{T_1' T_2' X' \bar{T}_2 \bar{T}_1 \bar{X}}^* \quad (91)$$

$$\varrho_{\bar{T}_2 \bar{T}_1 T_1 T_2} = \int_{\bar{X}, X} \int_{\bar{T}_1', T_1'} \int_{\bar{T}_2', T_2'} \int_{X', \bar{X}'} \mathcal{M}_{X T_1' T_2' \bar{X}' \bar{T}_2 \bar{T}_1} \mathcal{M}_{T_1 T_2 X' \bar{T}_2' \bar{T}_1 \bar{X}}^* \quad (92)$$

where the ordering of the Grassmann measures are defined so that the coefficient tensors of $\varrho_{T_1 T_2 \bar{T}_2 \bar{T}_1}$ and $\varrho_{\bar{T}_2 \bar{T}_1 T_1 T_2}$ are given by $\rho_{(t_1 t_2)(\bar{t}_1 \bar{t}_2)}$ and $\rho'_{(t_1' t_2')(\bar{t}_1' \bar{t}_2')}$ in equations (80) and (81), respectively. Note that $\varrho_{T_1 T_2 \bar{T}_2 \bar{T}_1}$ and $\varrho_{\bar{T}_2 \bar{T}_1 T_1 T_2}$ are Grassmann-even since $\mathcal{M}_{X T_1 T_2 \bar{X}' \bar{T}_2' \bar{T}_1'}$ is Grassmann-even. As we have demonstrated in section 2.5, the SVD of these Grassmann reduced density matrices results in the SVD of their coefficient tensors as in equations (82) and (83). The coefficient tensors P and Q in equations (87) and (88) are then determined similarly with the normal HOTRG. Since resulting $\mathcal{P}$ and $\mathcal{Q}$ are Grassmann-even, no extra sign factor appears in calculating the right-hand side of equation (86). The Grassmann tensor $\mathcal{T}_n$ in equation (86) gives the coarse-grained Grassmann tensor network $\text{gTr}[\prod_{n' \in \Lambda} \mathcal{T}_{n'}]$ with the same ordering of Grassmann measures in equation (72), as is evident from equation (89). Therefore, we can iterate the coarse-graining transformation as in figure 7 even for the Grassmann tensor network, without any extra difficulty.

We have reviewed two types of TRG algorithms so far and both of them employ the local approximation based on the (HO)SVD to define the coarse-graining transformations. By constructing a coarse-graining transformation that includes not only the local tensor(s) but also the effects of the surrounding tensors (they are usually referred to as the environment), we can construct a more accurate transformation. This kind of improvement is called the second RG [29, 32, 144]. One of the other ways to improve the TRG algorithms is to remove the redundant loop structure in the tensor network [28]. This can be achieved by the TNR [20, 145]. We will see several TNR-type algorithms in section 4.

In addition to the TRG approach, many other algorithms perform approximate tensor contractions, and they are used according to their purpose and cost performance. For example, corner transfer matrix RG [26] and time-evolving block decimation [146] are widely used in the tensor network computations based on the Hamiltonian formalism. See [130] as a recent review.

2.7. Practical remarks

Here, we see how to carry out the SVD of Grassmann tensors in practical computations. We again consider the Grassmann-even tensor $\mathcal{O}_{\Phi\Psi}$ in equation (67) as an example. Since $\mathcal{O}_{\Phi\Psi}$ is Grassmann-even, the corresponding coefficient tensor O_{ij} takes a non-zero value only when $\Phi^i \Psi^j$ is

Grassmann-even. Therefore, we can always convert O_{ij} into a block-diagonalized form by the matrix elementary operations; regarding i as a row and j as a column, O_{ij} can be

$$O = \begin{bmatrix} j: \text{even} & j: \text{odd} \\ O^{(\text{even})} & 0 \\ 0 & O^{(\text{odd})} \end{bmatrix} \begin{matrix} i: \text{even} \\ i: \text{odd} \end{matrix}, \quad (93)$$

where ‘ $i: \text{even}$ (odd)’ means the row index i such that Φ^i becomes Grassmann-even (odd). Similarly, ‘ $j: \text{even}$ (odd)’ means the column index j such that Ψ^j becomes Grassmann-even (odd). Using this basis, the SVD of O gives us

$$O_{ij} = \sum_{k,l} U_{ik} \sigma_{kl} V_{lj}^\dagger, \quad (94)$$

where U and $V^\dagger$ are unitary matrices such that

$$U = \begin{bmatrix} k: \text{even} & k: \text{odd} \\ U^{(\text{even})} & 0 \\ 0 & U^{(\text{odd})} \end{bmatrix} \begin{matrix} i: \text{even} \\ i: \text{odd} \end{matrix}, \quad (95)$$

$$V^\dagger = \begin{bmatrix} j: \text{even} & j: \text{odd} \\ V^{\dagger(\text{even})} & 0 \\ 0 & V^{\dagger(\text{odd})} \end{bmatrix} \begin{matrix} l: \text{even} \\ l: \text{odd} \end{matrix}, \quad (96)$$

and $\sigma_{kl} = \sigma_k \delta_{kl}$ with the singular value σ_k and Kronecker’s delta δ_{kl} in the form of

$$\sigma = \begin{bmatrix} l: \text{even} & l: \text{odd} \\ \sigma^{(\text{even})} & 0 \\ 0 & \sigma^{(\text{odd})} \end{bmatrix} \begin{matrix} k: \text{even} \\ k: \text{odd} \end{matrix}. \quad (97)$$

Recalling equations (32) and (69), we can associate k and l with new auxiliary Grassmann variables via Ξ^k and Ξ^l . In equations (95) and (97), ‘ $k: \text{even}$ (odd)’ should be understood as Ξ^k is of the Grassmann-even (odd) parity. In the same way, ‘ $l: \text{even}$ (odd)’ in equations (96) and (97) should be understood as Ξ^l is Grassmann-even (odd). Although we have two kinds of Grassmann variables Ξ^k and Ξ^l , Kronecker’s delta δ_{kl} in σ_{kl} enforces them to be of the same Grassmann parity. Therefore, we can easily define the new Grassmann parity functions corresponding to the new auxiliary Grassmann variables.

In the practical TRG computations, the Grassmann parity functions play a crucial role in restoring the Grassmann calculus on your computer. In the case of the Levin–Nave TRG explained in section 2.5, the data that should be kept on the computer are the coefficient tensor $T_{n;xtx't'}$ and Grassmann parity functions for each subscript. Here, we repeat the discussion from equation (73) to equation (76). The two kinds of SVD for the Grassmann tensors in equations (73) and (74) correspond to

$$T_{n;xtx't'} \simeq \sum_{a=1}^{D_{\text{LNTRG}}} A_{n;xta} B_{n;ax't'}, \quad (98)$$

$$(-1)^{f_x'(f_x+f_i)+f_x f_i} T_{n;xtx't'} \simeq \sum_{a=1}^{D_{\text{LNTRG}}} C_{n;xt'a} D_{n;ax't'}, \quad (99)$$

where A , B , C and D are defined in the similar way to equations (63) and (64). Notice that the extra sign factor in equation (99) comes from the rearrangement of auxiliary Grassmann variables. Using the block-diagonalized basis, one can immediately find the corresponding Grassmann parity for the new index a . Equation (75), or the diagram in figure 6, is now ready to be computed. We begin with contracting $\mathcal{A}$ and $\mathcal{D}$, that gives us an intermediate Grassmann tensor,

$$(\mathcal{AD})_{T_1 T \bar{X} T_2} = \int_{\bar{X}_1, X_1} \mathcal{A}_{n; X_1 T_1 T} \mathcal{D}_{n+\hat{1}; \bar{X} \bar{X}_1 T_2}, \quad (100)$$

which is described by the corresponding coefficient tensors as

$$(\mathcal{AD})_{t_1 t x' t_2} = \sum_{x_1, x'_1} (-1)^{f_{x_1}(f_{t_1}+f_{t_2})+f_{x'_1}f_{x'}} \mathcal{A}_{n; x_1 t_1 t} \delta_{x_1 x'_1} \mathcal{D}_{n+\hat{1}; x' x'_1 t_2}. \quad (101)$$

The contraction between $\mathcal{C}$ and $\mathcal{B}$ is

$$(\mathcal{CB})_{\bar{T}_1 X \bar{T} \bar{T}_2} = \int_{\bar{X}_2, X_2} \mathcal{C}_{n+\hat{2}; X_2 \bar{T}_1 X} \mathcal{B}_{n+\hat{1}+\hat{2}; \bar{T} \bar{X}_2 \bar{T}_2}, \quad (102)$$

which corresponds with

$$(\mathcal{CB})_{t'_1 x' t'_2} = \sum_{x_2, x'_2} (-1)^{f_{x_2}(f_{t'_1}+f_{x_2})+f_{x'_2}f_{t'_2}} \mathcal{C}_{n+\hat{2}; x_2 t'_1 x} \delta_{x_2 x'_2} \mathcal{B}_{n+\hat{1}+\hat{2}; t' x'_2 t'_2}. \quad (103)$$

Finally, the contraction between $(\mathcal{AD})$ and $(\mathcal{CB})$ gives a new Grassmann tensor in equation (75);

$$\mathcal{T}_{n'; XT \bar{X} \bar{T}} = \int_{\bar{T}_1, T_1} \int_{\bar{T}_2, T_2} (\mathcal{AD})_{T_1 T \bar{X} T_2} (\mathcal{CB})_{\bar{T}_1 X \bar{T} \bar{T}_2}, \quad (104)$$

which is restored by

$$\begin{aligned} T_{n'; x t x' t'} &= (-1)^{f_x(f_t+f_{x'})} \sum_{t_1, t'_1} \sum_{t_2, t'_2} (-1)^{f_{t_1}(f_t+f_{x'})+f_{t'_1}(f_{t'_1}+f_{x'})} \\ &\times (\mathcal{AD})_{t_1 t x' t_2} \delta_{t_1 t'_1} \delta_{t_2 t'_2} (\mathcal{CB})_{t'_1 x' t'_2}, \end{aligned} \quad (105)$$

where the sign factor inside the summations originates from reordering auxiliary Grassmann variables in $(\mathcal{AD})$ and $(\mathcal{CB})$. The sign factor outside the summations comes from reordering auxiliary Grassmann variables after the summations⁶.

The Grassmann tensor trace is also described by the coefficient tensor and parity functions. Suppose $\mathcal{T}_{n; XT \bar{X} \bar{T}}^{(N)}$ be a Grassmann tensor obtained by N times of Levin–Nave TRG or HOTRG transformation. The path integral defined on a lattice with 2^N sites is approximately given by $\text{gTr}[\mathcal{T}_n^{(N)}]$ and one finds

$$\text{gTr}[\mathcal{T}_n^{(N)}] = \sum_{x, t} (-1)^{f_x f_t} T_{n; x t x t}^{(N)}. \quad (106)$$

In the case of the finite-temperature system, we need to impose the anti-periodic boundary condition for the temporal direction, which can be done by using the Grassmann parity function f_t via

$$\text{gTr}[\mathcal{T}_n^{(N)}] = \sum_{x, t} (-1)^{f_x f_t + f_t} T_{n; x t x t}^{(N)}. \quad (107)$$

Note that the extra sign factor $(-1)^{f_t}$ is a result of replacing T^t with $(-T)^t$ in $\mathcal{T}_{n; XT \bar{X} \bar{T}}^{(N)}$.

3. Examples of numerical calculations

3.1. Wilson–Majorana fermions

Since the two-dimensional classical Ising model is one of the simplest models for critical phenomena with spontaneous $\mathbb{Z}_2$ symmetry breaking, it has been used as a benchmark to validate various TRG methods. Here, we begin with considering a fermionic model that is fully equivalent to the Ising model to see the validity of the Grassmann TRG.

A simple example of classical action quadratic in Grassmann variables is given by the action of Wilson–Majorana fermions:

$$\begin{aligned} S &= \frac{1}{2} \sum_n \bar{\eta}(n) \left(m_\eta + \sum_{\mu=1}^2 \gamma_\mu \partial_\mu^S - \frac{1}{2} \sum_{\mu=1}^2 \partial_\mu \partial_\mu^* \right) \eta(n) \\ &+ \frac{1}{2} \sum_n \bar{\chi}(n) \left(m_\chi + \sum_{\mu=1}^2 \gamma_\mu \partial_\mu^S - \frac{1}{2} \sum_{\mu=1}^2 \partial_\mu \partial_\mu^* \right) \chi(n) \\ &+ \sum_n \bar{\eta}(n) \left(\gamma_1 \partial_1^S - \gamma_2 \partial_2^S - \frac{1}{2} \partial_1 \partial_1^* + \frac{1}{2} \partial_2 \partial_2^* \right) \chi(n) \end{aligned} \quad (108)$$

with two-component Majorana spinors $\eta \equiv (\eta_1, \eta_2)^T$ and $\chi \equiv (\chi_1, \chi_2)^T$. The forward, the backward, and the symmetric difference operators are defined by ∂ , ∂^* , and $\partial^S = (\partial + \partial^*)/2$, respectively.

An equivalence between the lattice action for Wilson–Majorana fermions and the Ising model has been shown for two-dimensional honeycomb and square lattices for any choice of the periodic/anti-periodic boundary conditions [147]. The masses of the fermions are functions of the ‘reverse temperature’ κ :

$$m_\eta = \frac{2}{\kappa} (\sqrt{2} - 1 - \kappa), \quad m_\chi = -\frac{2}{\kappa} (\sqrt{2} + 1 + \kappa). \quad (109)$$

The critical point of the system $\kappa_c = \sqrt{2} - 1$ is associated with vanishing m_η , and this value is related to the critical point of the two-dimensional Ising model on a square lattice $\beta_c = \tanh^{-1} \kappa_c$. Even in such a model where there are several species of fermions, one can follow the prescription given in the previous section to derive a tensor network representation.

One can see the equivalence to the Ising model from the specific heat of this model in figure 8. Although we omit to

⁶ One can see that the algorithm should result in the $\mathbb{Z}_2$ -symmetric Levin–Nave TRG when we set all the Grassmann parity functions to zero.

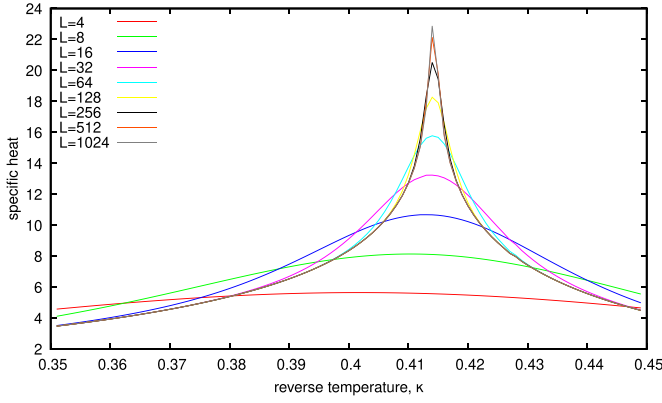


Figure 8. Specific heats of the Majorana–Wilson fermion system on several sizes of lattice. The specific heat is defined as the second derivative of free energy that is calculated by the Grassmann TRG with $D_{\text{LNTRG}} = 64$.

show a detailed finite-size scaling analysis, one can see the logarithmic divergence of the specific heat at the critical point $\kappa_c = \sqrt{2} - 1$.

In the latter section, we will show the ‘renormalization flow’ of this model given by the plain and the improved coarse-graining methods.

3.2. The Schwinger model

The Schwinger model, which is two-dimensional QED, is the most famous toy model for four-dimensional QCD. This is because the Schwinger model has similar properties to QCD, such as fermion confinement and chiral symmetry breaking. A θ term can also be introduced in the massive model, which undergoes a phase transition at $\theta = \pi$. The non-trivial effect of θ term in QCD is related to the strong CP problem. In addition, the massless model is exactly solvable and the mass perturbation is a useful technique to investigate the model with sufficiently small finite mass. In the HEP community, Shimizu and Kuramashi did a series of pioneering works for the two-dimensional Schwinger model with Wilson fermions [117–119].

In [117], where the Grassmann TRG was firstly applied to a lattice gauge theory, the Schwinger model

$$\begin{aligned}
 S = & -\frac{1}{2} \sum_n \sum_{\mu=1}^2 \bar{\psi}(n) \{ (1 - \gamma_\mu) U_\mu(n) \psi(n + \hat{\mu}) \\
 & + (1 + \gamma_\mu) U_\mu^\dagger(n - \hat{\mu}) \psi(n - \hat{\mu}) \} \\
 & + \frac{1}{2\kappa} \sum_n \bar{\psi}(n) \psi(n) - \beta \sum_n \cos(A_1(n) + A_2(n + \hat{1}) \\
 & - A_1(n + \hat{2}) - A_2(n)), \quad (110)
 \end{aligned}$$

where A is the phase of $U(1)$ link variable, was analyzed for some choices of the reverse coupling constant β : the strong coupling limit $\beta = 0$ and finite couplings $\beta = 5, 10$.

They confirmed their formulation and code by checking convergences of relative errors of free energy along with

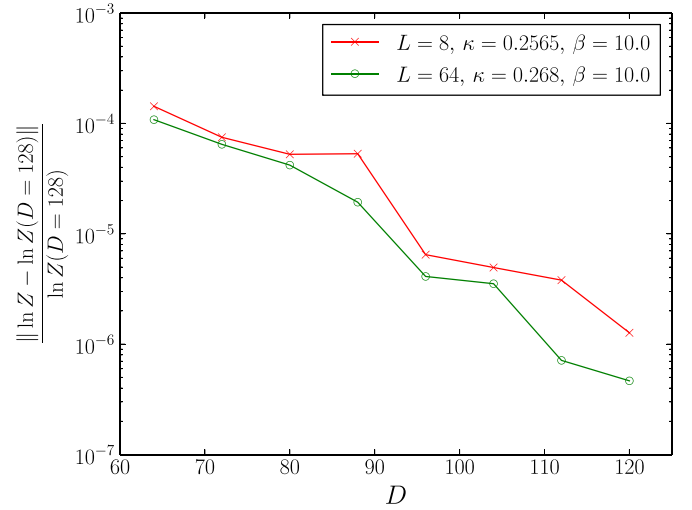


Figure 9. Adapted from [117]. Relative errors of the free energy. Reprinted (figure) with permission from [117], Copyright (2014) by the American Physical Society.

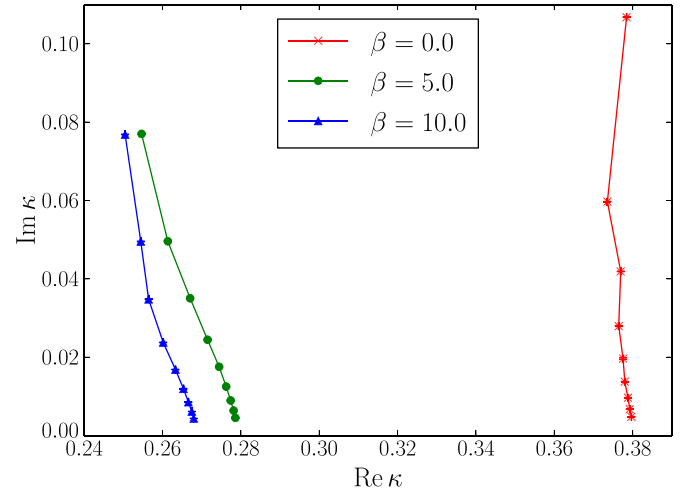


Figure 10. Adapted from [117]. Lee–Yang zeros in the complex κ -plane. Reprinted (figure) with permission from [117], Copyright (2014) by the American Physical Society.

increasing bond dimension (see figure 9). They defined the relative error by using the result obtained by a large bond dimension $D_{\text{LNTRG}} = 128$, while they fixed their bond dimension for the analyses to $D_{\text{LNTRG}} = 96$.

In the paper, finite-size scaling analyses is taken place for peak heights of the chiral susceptibility and the partition function zeros in the complex κ -plane. Using the Lee–Yang zeros, they did some reliable fittings to find out the critical exponents (see figure 10). Note that such an investigation in the complex parameter plane fully utilizes the absence of the sign problem for the TRG approach. Both in the strong coupling limit and for finite couplings, the critical exponents are shown to be the same as those of the two-dimensional Ising model. Moreover, the obtained critical mass (hopping parameter) is consistent with a previous work with the eight vertex model [148].

After this work, an analysis of the same model with the presence of θ term that impressed the efficiency of the TRG

approach on the community follows [118]. In that work, they studied the critical behavior on $\theta = \pi$ line, and in conclusion, on the line $\theta = \pi$, no phase transitions occur for $\kappa > \kappa_c$, a second order phase transition that belongs to the two-dimensional Ising universality class occurs at $\kappa = \kappa_c \simeq 0.2415$, and first-order phase transitions occur for $\kappa < \kappa_c$. This is exactly the expected result.

The Schwinger model with the topological term, where the staggered discretization for the fermions is applied, was studied by a different group [149]. In their paper, a special property of the model was mentioned: a tensor network representation can be constructed without having Grassmann variables on the network. Indeed this property was found in the context of Monte Carlo simulation in [150, 151].

4. Improved TRG methods for fermions

The Grassmann TRG methods introduced in sections 2.5 and 2.6 show remarkable performances as seen for some specific models in the latter sections. However, the key component of the RG methods was the (HO)SVD that gives the best approximation for local tensors [152] rather than the whole network. As we mentioned in the introduction, there are possible issues to describe the RG flows near critical points [27, 28, 99]. This is an important motivation to seek refined RG techniques. For tensor networks, the quality of the RG methods is often related to the ‘entanglement’ of the system. In [20], an improved renormalization technique where (dis)entanglers are introduced to the network and are variationally tuned so that the short-range correlation is removed at each RG step was introduced. Also, some new and less computationally demanding approaches like the loop-TNR [100] and the gilt-TNR [101] were developed. Such approaches are called TNR. Basically, the TNR approaches require some optimization steps on the network, but one can easily adapt them to the Grassmann tensor networks since the treatment of the Grassmann variables is exact and factored out from the bosonic part of the network. Note that, later in this section, we also show the Grassmann version of the bond-weighted TRG (BTRG), which represents a different notion than the TNRs.

4.1. CDL structure on tensor network

The difficulty in approaching critical points with the TRG has been illustrated by the CDL picture [28]. As a typical example, one can easily show that a toy tensor network that consists of the CDL tensor

$$T_{ijkl}^{\text{CDL}} = \Lambda_{i_1 l_2} \Lambda_{j_1 i_2} \Lambda_{k_1 j_2} \Lambda_{l_1 k_2}, \quad (111)$$

where Λ can be assumed to be a diagonal matrix for simplicity, is a fixed point of the TRG [28] (see figures 11 and 12). This means the usual TRG leaves short-range correlations in the network under each blocking step and this causes a loss of accuracy.

4.2. Removal of CDL from network

The TNR approaches mentioned at the beginning of this section attempt to remove the CDLs from the network. Some of the authors of this review article has checked the efficiency of the loop-TNR and the gilt-TNR for the Majorana–Wilson fermion system seen in section 3.1 and for the two-flavor Gross–Neveu model [120]. While the loop-TNR comes first in the chronological order, we show here how the gilt-TNR removes the CDL from a network for graphical simplicity.

In the gilt-TNR, recursive optimization steps are taken place to remove a CDL on a plaquette. To illustrate this, one can consider an SVD of the plaquette (see figure 13), which can be seen as an environment of a link. An important point here is that, by considering the link to be open⁷, the internal (i.e. the CDL) loop is seen to be not enclosed in the plaquette and connected to and only to the open link. With this logic, the CDL is captured by the unitary matrix ($\mathcal{U}$ in the figure) through the SVD. After the decomposition, the unitary matrix $\mathcal{U}$ is replaced by another one according to the ‘environment spectrum’ $\mathcal{S}$ so that the CDL loop will be truncated down.

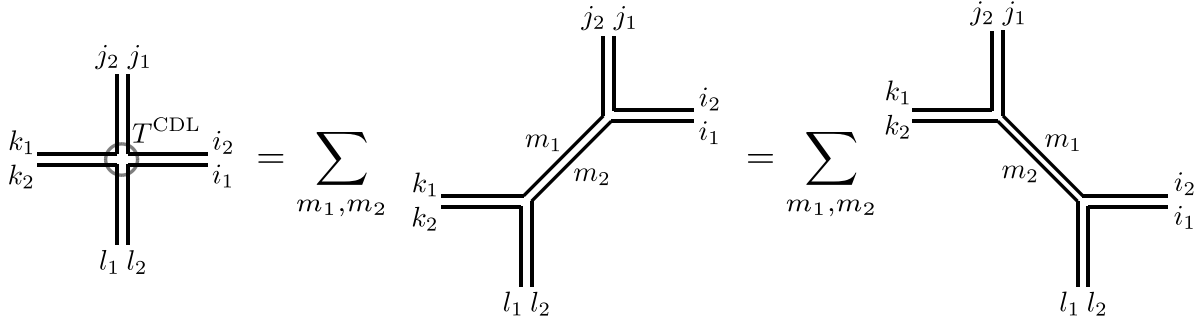
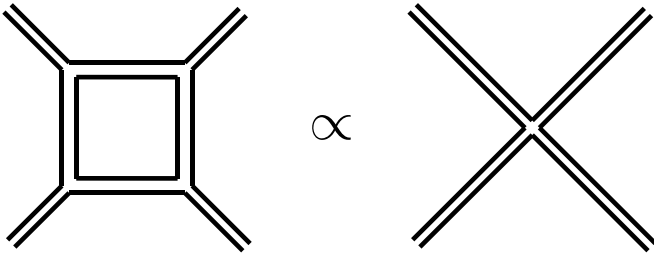
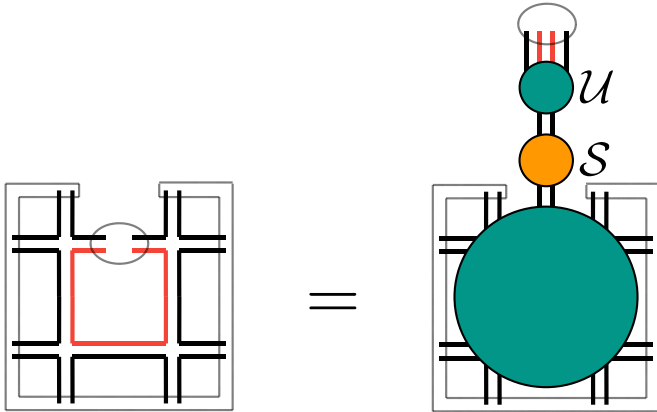
While the gilt-TNR is based on local replacements of tensor legs, the loop-TNR consists of an entanglement filtering gauge transformation of tensors and an optimization step that minimizes a bit global cost function compared to the usual TRG. One can see how is the CDL eliminated by the gauge transformation in the original paper [100], and see [28], where the notion of ‘entanglement filtering’ firstly appeared.

In [120], the performance of the Grassmann version of loop-TNR and gilt-TNR was inspected by computing the free energy and the determination of Fisher’s zeros. In the following, we review the RG flow of the singular values that are believed to store the information of the system.

The RG flow obtained by the plain Grassmann TRG and the loop-TNR are shown in figures 14 and 15⁸. The vertical and the horizontal axes represent the normalized singular value and how many iterations were taken place before that, respectively. Under these iterative coarse-graining algorithms, the space-time volume of the system grows rapidly along with the number of iterations; schematically it grows twice at each coarse-graining step. In figure 14 one cannot clearly distinguish the three panels where the reverse temperature is set to $0.9999\kappa_c$, κ_c , and $1.0001\kappa_c$. This is on account of a contamination by short-range information that is difficult to properly remove by the normal Grassmann TRG algorithm. On the other hand, the loop-TNR shows distinguishable fixed point structures at off-critical points such as $0.9999\kappa_c$, $1.0001\kappa_c$. Also, at the criticality κ_c , a scale-invariant structure is observed that clearly shows the superiority of the improved renormalization algorithm. Note that these behaviors of the singular values are qualitatively the same as those observed for the two-dimensional Ising model; the equivalence between the Ising model and the Majorana–Wilson fermion system can be seen in a sense like

⁷ Of course, the open link is to be closed after all.

⁸ In the reference, the result of the gilt-TNR is also shown; however, we omit to show it since the characteristic behavior is quite similar to that of the loop-TNR.

**Figure 11.** CDL tensor and decompositions.**Figure 12.** Coarse-graining of CDL tensor network leads to a CDL tensor network.**Figure 13.** SVD of plaquette. After this decomposition CDL loop is inside $\mathcal{U}$.

this. For the growth of the entanglement entropy and relationship to the Calabrese–Cardy formula, we refer the reader to the discussion in [120].

4.3. Bond-weighting technique

TRG algorithm can be improved by removing the short-range correlations represented by the loop entanglement. Several algorithms are proposed to remove these short-range correlations and achieve much higher accuracy than the original TRG algorithm even at the criticality as demonstrated above. At the same time, however, these algorithms usually require more computational cost than the original TRG. Recently, Adachi, Okubo, and Todo have proposed a new idea to improve the accuracy of TRG algorithms without increasing

the computational cost [153]. Their idea is to introduce some weight on the edge of the tensor network and construct the coarse-graining transformation including these weights. The algorithm is referred to as the BTRG, which can be regarded as a generalization of the Levin–Nave TRG. The advantage of the BTRG over the Levin–Nave TRG is demonstrated by benchmarking with the two-dimensional Ising model in [153]. The authors show that the BTRG outperforms the Levin–Nave TRG and the HOTRG at the same bond dimension. Both for the BTRG and the Levin–Nave TRG, their computational times are proportional to $O(D^5)$ and their memory footprints are $O(D^3)$. In the two-dimensional HOTRG, the computational time scales with $O(D^7)$, and the scaling of the memory cost is $O(D^4)$. Therefore, the BTRG has shown the best performance among these three algorithms. Moreover, the authors numerically demonstrate that non-trivial fixed point tensors can be constructed in the thermodynamic limit by the BTRG. Recently, the BTRG has been applied to investigate the phase structure of the $CP(1)$ model with a topological θ term [154].

We review the BTRG algorithm for the two-dimensional square tensor network which is generated by a four-leg fundamental tensor. The algorithm is schematically explained in figure 16. The key step in the BTRG is the low-rank approximation of the four-leg tensor based on the SVD with a hyperparameter such as

$$T_{abcd} \simeq \sum_{i=1}^D U_{abi} \sigma_i^{(1-k)/2} \sigma_i^k \sigma_i^{(1-k)/2} V_{icd}^\dagger, \quad (112)$$

where $k \in \mathbb{R}$ denotes the hyperparameter. If we set $k = 0$, equation (112) exactly corresponds to the tensor decomposition employed in the Levin–Nave TRG as shown in equations (63) and (64). With $k \neq 0$, we obtain an extra factor σ_i^k , which is regarded as a weight on the bond i . The authors in [153] have given a stationary condition equation,

$$\left[\sigma_i^{(1-k)/2} \right]^4 \left[\sigma_i^k \right]^4 = \sigma_i, \quad (113)$$

that determines the optimal choice of the hyperparameter k . Equation (113) enforces the singular value spectrum to be invariant under the sequential coarse-graining procedure in the BTRG, assuming that the local four-leg tensors and the bond weights converge after sufficiently many times coarse-graining transformations and the matrices U and V do not

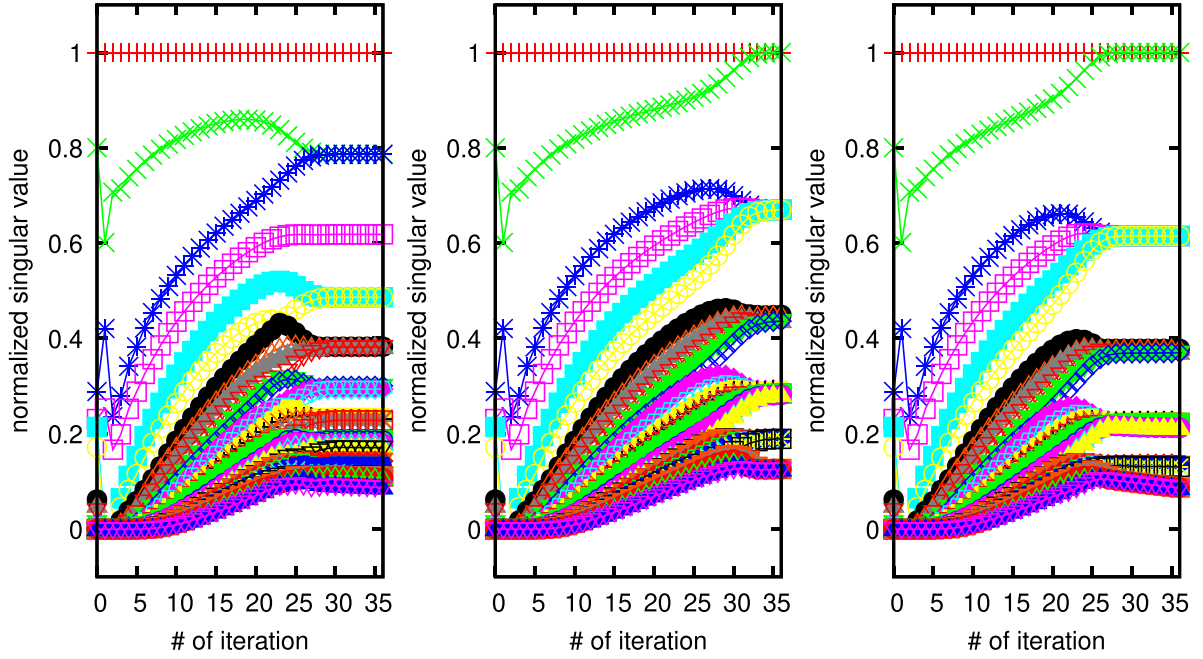


Figure 14. Normalized singular values produced by Grassmann TRG at $\kappa = 0.9999\kappa_c$ (left), $\kappa = \kappa_c$ (middle), and $\kappa = 1.0001\kappa_c$ (right). The bond dimension is set to 64.

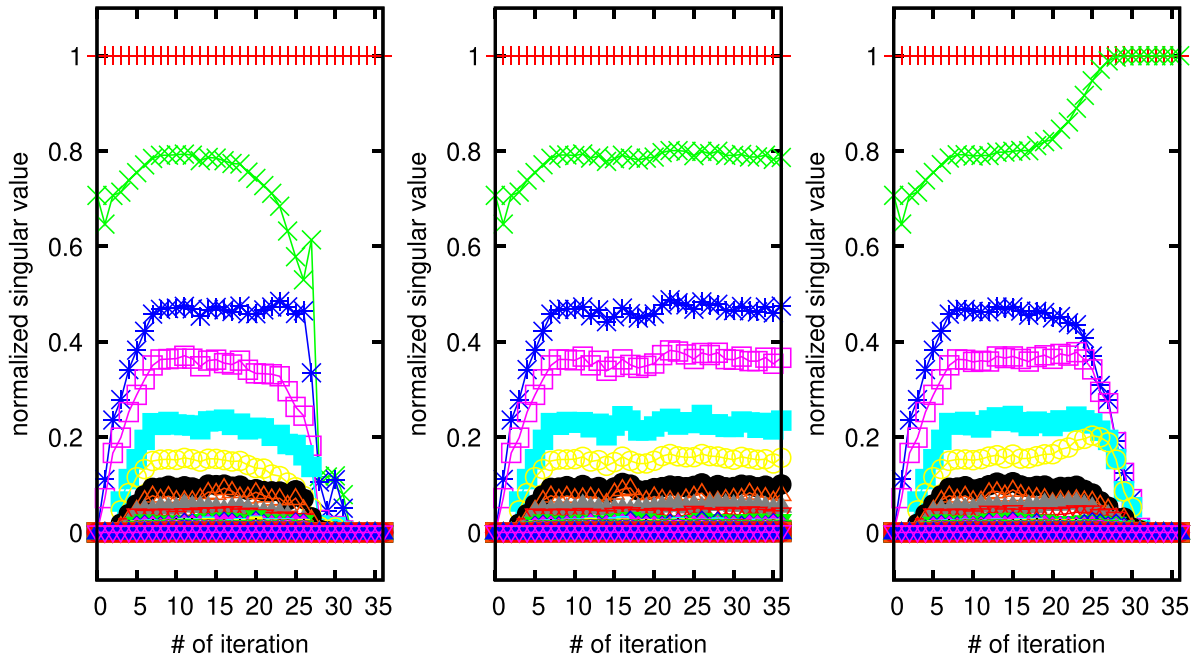


Figure 15. Normalized singular values produced by Grassmann loop-TNR at $\kappa = 0.9999\kappa_c$ (left), $\kappa = \kappa_c$ (middle), and $\kappa = 1.0001\kappa_c$ (right). The bond dimension is set to 16.

affect the spectrum. According to equation (113), the optimal value of the hyperparameter for the two-dimensional square tensor network should be $k = -1/2$, which has been numerically confirmed in [153]. We note that the application of the bond-weighting technique to other TRG algorithms has been discussed in [155].

It should be emphasized that this derivation is completely independent of the details of the lattice theory. Actually, [126]

shows that the bond-weighting method improves the accuracy of the Grassmann TRG. Figure 17 shows the relative error of the free energy for the two-dimensional free massless Wilson fermion. With the fixed bond dimension, the bond-weighting method always achieves higher accuracy compared with the normal TRG. Notice that both computations require the same computational complexity at the same bond dimension. A sample implementation of the BTRG for the Gross–Neveu

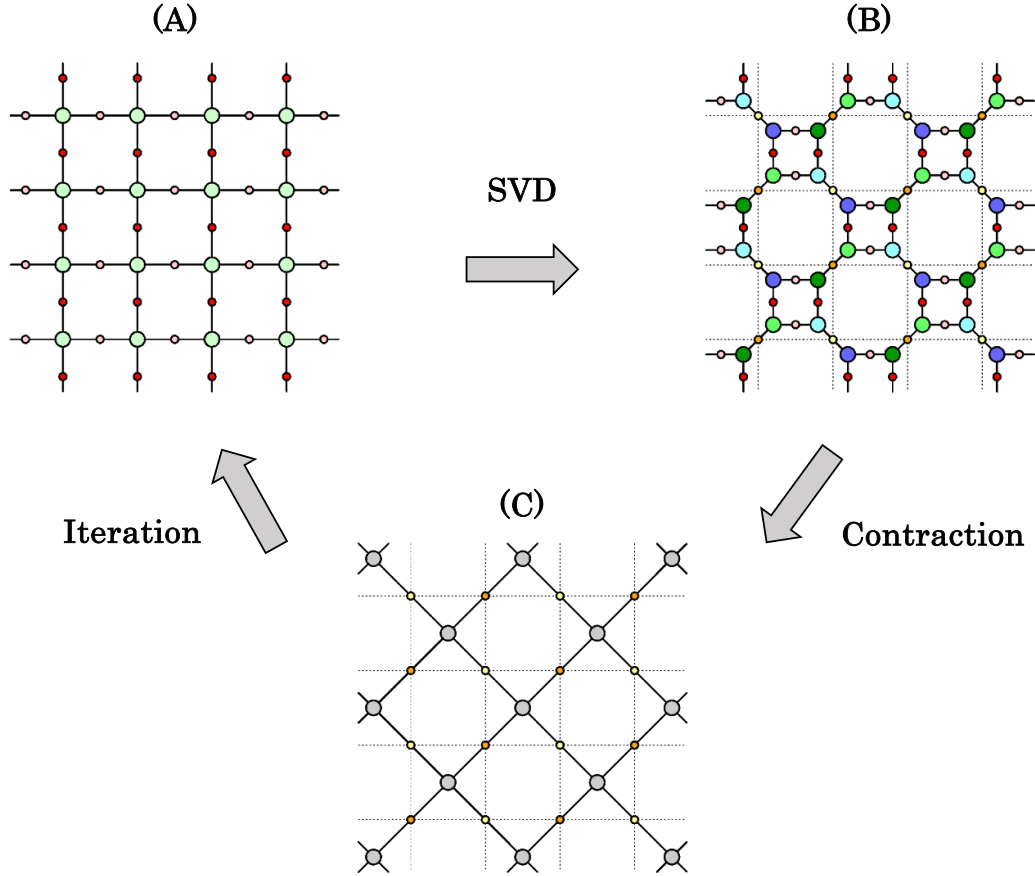


Figure 16. Schematic illustration of the BTRG algorithm. Background dotted lines denote a real-space square lattice. (A) Initial tensor network with bond weights on the lattice. (B) SVD introduces three-leg tensors and new bond weights. (C) New tensor network with bond weights by contracting four types of three-leg tensors and four bond weights.

model with Wilson fermions at finite density is shown in [128], whose web documentation is also provided⁹. Using the code, one can reproduce the result shown in figure 17.

4.4. Multilayered tensor network formulations for N_f -flavor fermions

There is no difficulty in expressing the path integral of the lattice fermion system as the Grassmann tensor network with N_f flavors. In practice, however, the size of the resulting Grassmann tensor scales exponentially for N_f and a $O(e_f^N)$ computational memory is required in the numerical computations. This issue has been started to be addressed recently by Akiyama [103] and also by Yosprakob *et al* [127].

Akiyama [103] has employed the matrix product decomposition (MPD) to introduce a virtual direction so that each flavor degree of freedom is assigned to the different layers orthogonal to the virtual direction. MPD is a common idea in the tensor network methods such as the MPS and matrix product operator [156, 157]. Akiyama [103] has particularly utilized a canonical form of the MPD proposed in [158]. Thanks to the MPD, the memory cost for each local Grassmann tensor

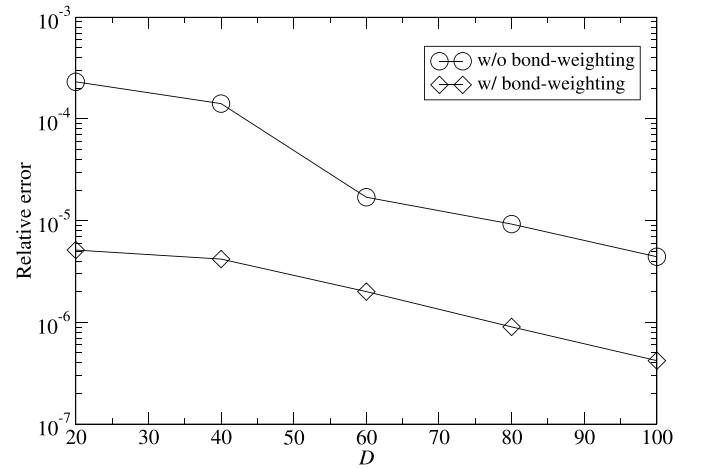


Figure 17. Relative error of the free energy for the two-dimensional free massless Wilson fermion. The hyperparameter is set at $k = -1/2$.

is reduced from $O(e_f^N)$ to $O(N_f)$, and the technique has been benchmarked with the two-dimensional Gross–Neveu model at finite density with the $N_f = 2, 3$ Wilson fermions. Although a naive formulation provides the two-dimensional Grassmann

⁹ <https://github.com/akiyama-es/Grassmann-BTRG>.

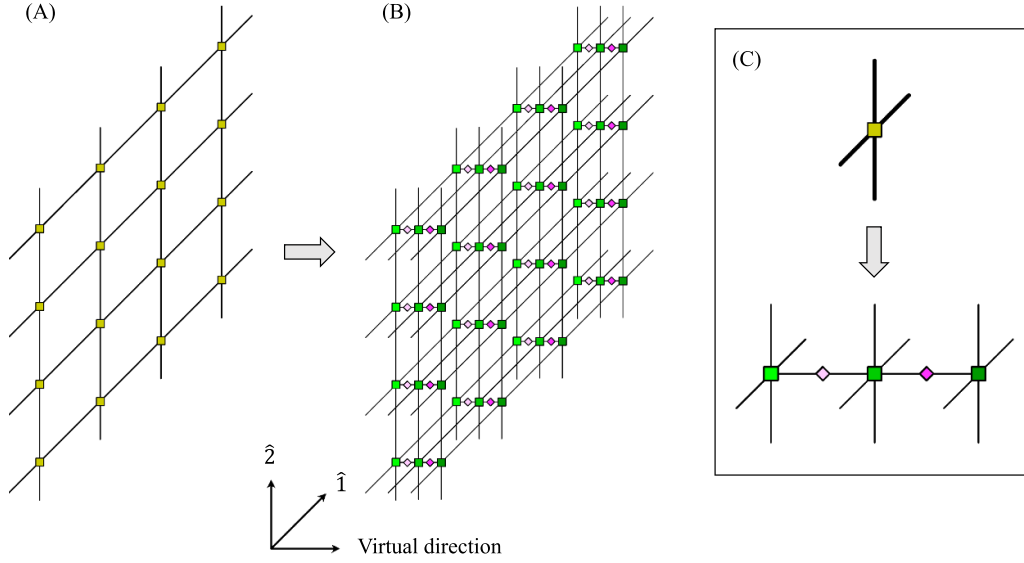


Figure 18. Adapted from [103]. Multilayered Grassmann tensor network representation for the path integral of the two-dimensional $N_f = 3$ Gross–Neveu model with Wilson fermions. (A) Original Grassmann tensor network representation defined on the square lattice. (B) Three-layered Grassmann tensor network. (C) MPD rewrites the original fundamental tensor (yellow) with three fundamental tensors (green) and two kinds of singular values (purple). Reproduced from [103]. CC BY 4.0.

tensor network whose bond dimension equals 4^{N_f} , the MPD alternatively gives the $(2 + 1)$ -dimensional network, N_f sites along the virtual direction, with the bond dimension four without any approximation. The schematic picture is shown in figure 18. Based on the latter representation, the Silver Blaze phenomenon in the pressure and number density is reproduced with a relatively small bond dimension.

Yosprakob *et al* [127] has proposed a compression scheme for the initial tensor representation of lattice gauge theories with the N_f -flavor fermions. They have introduced replicas of the original gauge field, which allows one to separate the local Grassmann tensor into multiple layers associated with the fermion flavor. Based on this description, each layer is individually compressed by the isometry insertion; In the case of the $\mathbb{Z}_K$ gauge theory with the N_f -flavor Wilson fermions, the original tensor whose size is $16^4 K^{10}$ is converted into the compressed one with $D^4 K^2$ elements, where D is the bond dimension introduced by the isometries. Even with $D = 8$, the difference between the resulting $\ln Z$ at finite gauge coupling and chemical potential is suppressed less than $O(10^{-15})$. Since gauge fields are replicated by the Kronecker deltas, they are maximally entangled along the flavor direction. Therefore, the tensor contractions along the flavor direction are carried out before the two-dimensional spacetime coarse-graining is performed. Using this technique, the chiral susceptibility of the two-dimensional infinite-coupling $\mathbb{Z}_2$, $\mathbb{Z}_4$, and $U(1)$ gauge theories with the $N_f = 1, 2$ Wilson has been computed. The critical hopping parameters have been determined for each case. The pressure and number density as functions of the chemical potential have also been provided in the case of $\mathbb{Z}_2$ gauge theory with $N_f = 1, 2, 4$ at finite gauge coupling, where the Silver Blaze phenomenon has been successfully captured.

In addition, Yosprakob has been recently developing a Python package for the Grassmann tensor network

computations in [129], whose web documentation is also provided¹⁰. The Schwinger model is implemented as an example.

5. Relativistic models with fermion interactions

5.1. Gross–Neveu model

The Gross–Neveu model is a well-known toy model for QCD. Since it shares several important features of the QCD such as asymptotic freedom and a dynamical mass generation mechanism via symmetry breaking, the model is a good test bed for new computational methods. Here, we consider the model defined with Wilson fermions. The lattice action reads

$$\begin{aligned}
 S = & -\frac{1}{2} \sum_{n, \nu} \sum_{f=1}^{N_f} \left\{ e^{\mu \delta_{\nu, 2}} \bar{\psi}^{(f)}(n) (r \mathbb{1} - \gamma_\nu) \psi^{(f)}(n + \hat{\nu}) \right. \\
 & \left. + e^{-\mu \delta_{\nu, 2}} \bar{\psi}^{(f)}(n + \hat{\nu}) (r \mathbb{1} + \gamma_\nu) \psi^{(f)}(n) \right\} \\
 & + \sum_n \sum_{f=1}^{N_f} (m + 2r) \bar{\psi}^{(f)}(n) \psi^{(f)}(n) \\
 & - \frac{g_\sigma^2}{2N_f} \sum_n \left(\sum_f \bar{\psi}^{(f)}(n) \psi^{(f)}(n) \right)^2 \\
 & - \frac{g_\pi^2}{2N_f} \sum_n \left(\sum_f \bar{\psi}^{(f)}(n) i \gamma_5 \psi^{(f)}(n) \right)^2, \quad (114)
 \end{aligned}$$

where $\psi^{(f)}(n)$ and $\bar{\psi}^{(f)}(n)$ are the Wilson fermions with the flavor index f . g_σ^2 and g_π^2 are the four-fermi coupling constants

¹⁰ <https://ayosprakob.github.io/grassmannntn/>.

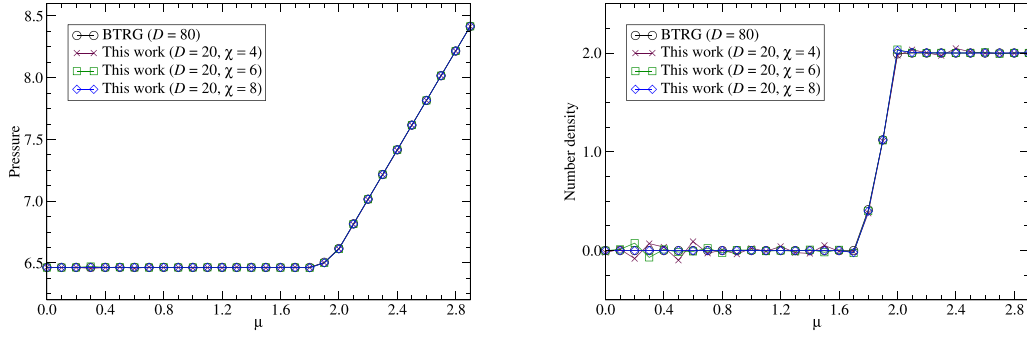


Figure 19. Adapted from [103]. Pressure (left) and number density (right) of the $N_f = 2$ Gross–Neveu model with Wilson fermions. D denotes the bond dimension for spacetime indices and χ does the bond dimension along the virtual direction in the multilayered network. $m = 1$ and $g^2 = 10$. Reproduced from [103]. CC BY 4.0.

and m , μ , and r represent mass, chemical potential, and the Wilson parameter, respectively.

Takeda and Yoshimura studied the model with $N_f = 1$ at finite density in [102], which is the first application of the Grassmann TRG to the finite-density system. They employed the Levin–Nave TRG algorithm with the bond dimension up to $D_{\text{LNTRG}} = 64$ to compute the fermion number density and its susceptibility as functions of μ . They successfully observed that the number density saturated to one with sufficiently large chemical potential. In addition, they considered the model on an anisotropic finite lattice with $(N_1, N_2) = (64, 32), (96, 32)$, where N_1 and N_2 denote the spatial and temporal lattice sizes respectively, and found that there were two peaks in the susceptibility under the presence of the finite four-fermi coupling. Throughout their analysis, they pointed out that the finite bond dimension effect could be enhanced not only at critical points but also in crossover regions.

Recently, Akiyama has investigated the model with $N_f = 2, 3$ at finite density [103]. The pressure and number density on a square lattice were computed as functions of chemical potential using two methods, the bond-weighting method and multilayered formulation as shown in figures 19 and 20. In the zero-temperature limit, we observe the Silver Blaze phenomena, where the thermodynamic quantities show no dependence on μ as long as μ is smaller than the mass of the lightest excitation. The results obtained by the two methods are consistent and the number density saturates to two and three for $N_f = 2, 3$, respectively. They noted that the finite bond dimension effect could be enhanced in the Silver-Blaze regime in the multilayered formulation.

5.2. QCD in the infinite-coupling limit

Bloch and Lohmayer made the first study of QCD in the infinite-coupling limit with the Grassmann TRG approach [104]. The lattice action with the staggered quarks is given by

$$S = \sum_n \left\{ \eta_1(n) \gamma \bar{\psi}(n) \left[e^\mu U_1(n) \psi(n+\hat{1}) - e^\mu U_1^\dagger(n-\hat{1}) \psi(n-\hat{1}) \right] + \eta_2(n) \bar{\psi}(n) \left[U_2(n) \psi(n+\hat{2}) - U_2^\dagger(n-\hat{2}) \psi(n-\hat{2}) \right] + 2m \bar{\psi}(n) \psi(n) \right\}, \quad (115)$$

where m and μ denote the mass and chemical potential. The lattice site is labelled by $n = (n_1, n_2)$ with the temporal coordinate n_1 and spatial coordinate n_2 . The action has an anisotropy factor γ in the temporal hopping terms. The link variables $U_\nu(n)$ take their value on $SU(3)$ and $\psi(n)$ and $\bar{\psi}(n)$ are the staggered quark fields described by the three-component Grassmann variables. The staggered sign function is defined via $\eta_1(n) = 1$ and $\eta_2(n) = (-1)^{n_1}$. The (anti-)periodic boundary condition is assumed in the spatial (temporal) direction. The authors integrate out the $SU(3)$ link variables exactly as shown in [159, 160] and the path integral is rewritten by the mesonic and baryonic contributions, following [161], characterized by their occupation numbers. Note that this kind of dual formulation has also been applied in [162] to investigate the infinite-coupling $U(N)$ gauge theories with staggered fermions in three and four dimensions with the variant of HOTRG. Effectively reducing the number of configurations with vanishing contribution in the path integral, they obtain the Grassmann tensor network representation where the bosonic tensor is defined by a four-leg tensor and each subscript is of dimension six. The resulting Grassmann tensor network is approximately contracted by the HOTRG, where the authors have applied an improved method to derive the projectors in the algorithm [163].

The authors calculated the thermodynamic potential $\ln Z/V$ as a function of μ and m on 2^2 and 4^2 lattices and confirmed that the exact results were reproduced. Although the convergence for the bond dimension D_{HOTRG} becomes slower with the smaller m , it has been shown that the accuracy of the calculations is sufficient to take the chiral limit: a quadratic

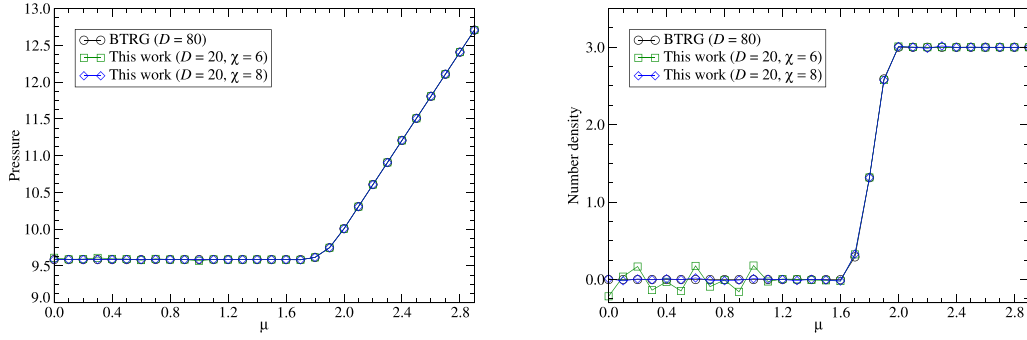


Figure 20. Adapted from [103]. Pressure (left) and number density (right) of the $N_f = 3$ Gross–Neveu model with Wilson fermions. D denotes the bond dimension for spacetime indices and χ does the bond dimension along the virtual direction in the multilayered network. $m = 1$ and $g^2 = 10$. Reproduced from [103]. CC BY 4.0.

fit in $1/D_{\text{HOTRG}}$ successfully extrapolates $\ln Z/V$ at $m = 0$ on a 1024^2 lattice to $D_{\text{HOTRG}} \rightarrow \infty$ using the results on $40 \leq D_{\text{HOTRG}} \leq 128$.

The authors then computed the chiral condensate $\langle \bar{\psi}\psi \rangle$ at vanishing chemical potential. They extrapolated $\langle \bar{\psi}\psi \rangle$ at finite mass to $D_{\text{HOTRG}} \rightarrow \infty$ before taking the infinite-volume limit. For $m \leq 0.005$, the condensate was nicely fitted by $\lim_{V \rightarrow \infty} \langle \bar{\psi}\psi \rangle = am^b$, with $a = 2.77$ and $b = 0.0414$. Therefore, it is confirmed that the chiral symmetry is not dynamically broken in infinite-coupling QCD with staggered quarks in two dimensions. They also studied the number density and chiral condensate at finite chemical potential with $m = 0.1$ and $D_{\text{HOTRG}} = 64$. In the infinite-volume limit, they have found that a first-order phase transition takes place at $\mu_c \simeq 0.3508$. With $\mu > \mu_c$, they have observed that the number density saturates to three and the chiral symmetry is restored. Throughout the study, they employed the stabilized finite-difference method developed in [164] to calculate the chiral condensate and number density by the numerical differentiation.

5.3. NJL model

Akiyama, Kuramashi, Yamashita, and Yoshimura made the first application of the TRG approach to the four-dimensional lattice fermions [105]. They investigated the chiral phase transition in the NJL model at finite density. Since this model is an effective field theory of the QCD, the efficiency of the TRG approach for the NJL model should be addressed from the viewpoint of the future application of the TRG method toward the QCD at finite density. The model is defined with the staggered fermions by the following action,

$$S = \frac{1}{2} \sum_n \sum_{\nu=1}^4 \eta_{\nu}(n) \left[e^{\mu \delta_{\nu,4}} \bar{\chi}(n) \chi(n + \hat{\nu}) - e^{-\mu \delta_{\nu,4}} \bar{\chi}(n + \hat{\nu}) \chi(n) \right] + m \sum_n \bar{\chi}(n) \chi(n) - g_0 \sum_n \sum_{\nu} \bar{\chi}(n) \chi(n) \bar{\chi}(n + \hat{\nu}) \chi(n + \hat{\nu}), \quad (116)$$

where $\chi(n)$ and $\bar{\chi}(n)$ are the staggered fermions described by the single-component Grassmann variables. $\eta_{\nu}(n)$ is the staggered sign function $\eta_{\nu}(n) = (-1)^{n_1 + \dots + n_{\nu-1}}$

with $\eta_1(n) = 1$. m , g_0 , and μ represent the mass, four-fermi coupling constant, and chemical potential. When $m = 0$, equation (116) is invariant under the continuous transformation,

$$\chi(n) \rightarrow e^{i\alpha \eta_5(n)} \chi(n), \quad \bar{\chi}(n) \rightarrow \bar{\chi}(n) e^{i\alpha \eta_5(n)}. \quad (117)$$

This global symmetry is regarded as the chiral symmetry for the staggered NJL model.

Decomposing the hopping terms and four-fermi interaction term, the Grassmann tensor network representation for the path integral is available. Akiyama [103] introduces the eight kinds of local tensors to describe the Grassmann tensor network. This is because the staggered theory defined in equation (116) a little bit breaks the translational invariance on the lattice. Note that the site dependence appearing in the local tensor is characterized just by $\eta_{\nu}(n)$ so that the resulting Grassmann tensor network has a periodic structure. See references [103, 165] for the detailed derivation of the Grassmann tensor network representation.

The authors in [103] developed the anisotropic TRG (ATRG) algorithm [166] for fermions. The ATRG allows us to approximately contract d -dimensional tensor networks with the $O(D_{\text{ATRG}}^{2d+1})$ complexity and $O(D_{\text{ATRG}}^{d+1})$ memory cost. These costs should be compared with the $O(D_{\text{HOTRG}}^{4d-1})$ complexity and $O(D_{\text{HOTRG}}^{2d})$ memory cost in the HOTRG. This drastic cost reduction is a result of an additional approximation for fundamental tensors. The further cost reduction technique for the ATRG has been provided by Oba in [167]. In addition to the ATRG, several other algorithms have been proposed for the higher-dimensional systems [168, 169]. As a validation of the ATRG for fermionic models, they first considered the NJL model in the heavy-dense limit, where $m \rightarrow \infty$ and $\mu \rightarrow \infty$ keeping the ratio of e^{μ}/m fixed. In this limit, the model can be solved analytically [170]. The number density and fermion condensate as functions of μ in the thermodynamic and vanishing temperature limits were calculated by the ATRG with the bond dimension $D_{\text{ATRG}} \leq 30$ and the results showed a good agreement with the analytic ones.

The authors then studied the chiral phase transition in the cold and dense regime characterized by $\mu/T = O(10^3)$. Due to the sign problem, such a cold and dense regime is inaccessible with the standard Monte Carlo simulation. They enlarged

the bond dimension up to $D_{\text{ATRG}} = 55$ in the strong coupling regime, where they found a clear signal of the first-order transition in the chiral condensate, and the chiral symmetry was restored with $\mu > \mu_c$. This is exactly the predicted result by the mean-field theory [171] and the functional RG [172]. The authors also computed the pressure and number density which are fundamental ingredients in the equation of state. All the thermodynamic quantities obtained by the ATRG have shown that the model undergoes the first-order phase transition in the cold and dense region.

5.4. $\mathcal{N} = 1$ Wess–Zumino model

The interacting two-dimensional $\mathcal{N} = 1$ Wess–Zumino model, a simple supersymmetric model, shows a vanishing partition function (Witten index) [173]; i.e. This model suffers from a serious sign problem as in the case of other generic supersymmetric models. (See [174] for a review.)

The Euclidean continuum action of the model is defined by

$$S = \int d^2x \left\{ \frac{1}{2} (\partial_\mu \phi)^2 + \frac{1}{2} W'(\phi)^2 + \frac{1}{2} \bar{\psi} (\gamma_\mu \partial_\mu + W''(\phi)) \psi \right\}, \quad (118)$$

where ϕ and ψ are a one-component real scalar field and a two-component Majorana spinor field, respectively. The superpotential $W(\phi)$ is a function of ϕ and is the source of the Yukawa- and ϕ^n -interactions.

The Majorana condition for ψ is given by

$$\bar{\psi} = -\psi^T C^{-1} \quad (119)$$

with the charge conjugation matrix C ,

$$C^T = -C, \quad C^\dagger = C^{-1}, \quad C^{-1} \gamma_\mu C = -\gamma_\mu^T. \quad (120)$$

The continuum action above can be shown to be invariant under the supersymmetry transformation

$$\delta \phi = \bar{\epsilon} \psi, \quad (121)$$

$$\delta \psi = (\gamma_\mu \partial_\mu \phi - W'(\phi)) \epsilon, \quad (122)$$

where ϵ is a two-component Grassmann number that satisfies the Majorana condition (119).

Figure 21 shows the partition function of the free $\mathcal{N} = 1$ Wess–Zumino model, whose superpotential is defined by $W(\phi) = (1/2)m\phi^2$ with the mass parameter m , on $V = 2 \times 2$ lattice. (Note that the sign problem occurs even in this free case.) The periodic boundary conditions are imposed in all directions for both fermions and bosons. For this specific case, the analytical solution can be shown to be 1 for the $m > 0$ region shown in the figure. The Grassmann TRG results show better agreement with the exact solution at larger masses although smaller masses seem to be difficult. As for the difficulty in the small mass region, the authors of [106] concluded that it is due to the lack of damping factor in the local Boltzmann weight; in other words, the reason is that the Gauss–Hermite quadrature applied to the scalar boson part does not converge.

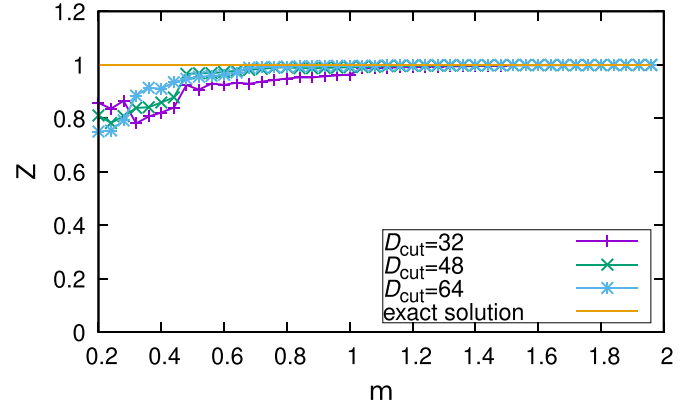


Figure 21. Adapted from [74]. The partition function of the $\mathcal{N} = 1$ free Wess–Zumino model as a function of m on $V = 2 \times 2$ lattice. Reprinted (figure) with permission from [74], Copyright (2022) by the American Physical Society.

Even though they showed results only for the non-interacting case, their construction of the tensor network does not depend on the shape of the superpotential, so that further studies of the interacting Wess–Zumino model are awaited.

5.5. Non-abelian lattice gauge theories coupled to fermions

Non-abelian gauge theories have huge internal degrees of freedom, and this fact prevents one from building a tensor network representation in a non-expensive way. Indeed, tensor network studies for gauge theories are limited to abelian cases when considering coupling to fermions. Recently a subgroup of the authors reported a way to construct a tensor network representation of non-abelian gauge theories with a reasonable numerical complexity [107]. In their work, parameterized group elements are discretized via a Gaussian quadrature method, and a further reduction of the degrees of freedom is achieved by applying the higher-order orthogonal iteration (HOOI) algorithm [175, 176] to plaquette tensors. Moreover, for the fermion part, they adopt the reduced staggered formulation [177] to completely eliminate the redundancies for the fermion part.

They numerically showed the accuracy of their approximation by checking the partition function and the average plaquette for the $SU(2)$ case. Surprisingly, the convergence of the HOOI algorithm is so fast, and the loss of accuracy by the truncation is shown to be quite mild. In other words, the main source of the error is the quadrature method with a few number of Gaussian nodes; they adopted up to 5 for the quadrature at each angle. It would be interesting to see how such a drastic approximation is tolerable for more complicated cases.

6. The Hubbard model

The Hubbard model is one of the most fundamental lattice fermion models describing the itinerant spin-1/2 electrons via the repulsive Coulomb interaction. The model is defined by the following Hamiltonian,

$$H = -t \sum_{\langle ij \rangle} \sum_{s=\uparrow, \downarrow} \left(c_{i,s}^\dagger c_{j,s} + c_{j,s}^\dagger c_{i,s} \right) + U \sum_{i \in \Lambda} n_{i,\uparrow} n_{i,\downarrow}. \quad (123)$$

The first term, the tight-binding Hamiltonian, represents the kinetic energy of electrons and the second one shows the repulsive Coulomb potential with $U \geq 0$. $\langle ij \rangle$ labels the nearest-neighbor sites in the d -dimensional hypercubic lattice Λ . The interesting aspect of the Hubbard model originates from the fact that each term is diagonalizable in a different space; the kinetic term is diagonalized in the momentum space but the Coulomb potential term is diagonalized in the real space. Exact solutions of the Hubbard model are available only in the cases of $d = 1$ [178] and $d = \infty$ [179, 180], despite its simplicity. In the condensed matter community, the model in general dimensions and on various lattice geometries have been extensively studied with mean-field approaches, field-theoretical ways, and numerical methods. We recommend for interested readers to see several recent reviews [108–114] and references therein.

Let us review the Hubbard model very briefly. The creation and annihilation operators $c_{i,s}^\dagger$ and $c_{i,s}$ satisfy the anti-commutation relation,

$$\begin{aligned} \{c_{i,s}, c_{i',s'}^\dagger\} &= \delta_{ii'} \delta_{ss'}, & \{c_{i,s}, c_{i',s'}\} &= 0, \\ \{c_{i,s}^\dagger, c_{i',s'}^\dagger\} &= 0. \end{aligned} \quad (124)$$

The number operator is defined via

$$N = \sum_{i,s} n_{i,s} \quad (125)$$

with $n_{i,s} = c_{i,s}^\dagger c_{i,s}$. The spin operator at the site i is given by $\mathbf{S}_i = (S_i^{(x)}, S_i^{(y)}, S_i^{(z)})$, where

$$S_i^{(\nu)} = \frac{1}{2} \sum_{s,s'} c_{i,s}^\dagger \sigma_{ss'}^{(\nu)} c_{i,s'}. \quad (126)$$

$\sigma^{(\nu)}$ ($\nu = x, y, z$) is the Pauli matrices. The total spin operator is

$$\mathbf{S} = \sum_i \mathbf{S}_i. \quad (127)$$

The Hamiltonian in equation (123) has the global $U(2) = U(1) \times SU(2)$ symmetry. With $R \in U(2)$, equation (123) is invariant under the transformation,

$$c_{i,s} \rightarrow \sum_{s'} R_{s,s'} c_{i,s'}. \quad (128)$$

This symmetry results in the global charge conservation and the spin isotropy. The global charge conservation originates from the global $U(1)$ symmetry and the total particle number N is a good quantum number. The spin isotropy is a result of the global $SU(2)$ symmetry and $S^{(z)}$ and $\mathbf{S}^2$ are good quantum numbers.

The grand canonical Hamiltonian is given by $H - \mu N$, where μ is the chemical potential. Since the model describes

the spin-1/2 electrons, $\langle N \rangle = 2|\Lambda|$ with the total lattice sites $|\Lambda|$. When the filling is a single electron per site, that is $\langle N \rangle = |\Lambda|$, the situation is referred to as half-filling. At half-filling, the system acquires the so-called particle-hole symmetry. To see this, let us consider the transformation such that

$$c_{i,s} \rightarrow \eta_i c_{i,s}^\dagger, \quad (129)$$

where $\eta_i = \pm 1$ on the even and odd lattice sites, respectively. The Hamiltonian in equation (123) is then transformed as

$$H \rightarrow H + U(|\Lambda| - N), \quad (130)$$

which means that equation (129) is the symmetry if the system is at half-filling $\langle N \rangle = |\Lambda|$. From the viewpoint of the grand canonical Hamiltonian,

$$H - \mu N \rightarrow H + U(|\Lambda| - N) + \mu N - 2\mu|\Lambda|, \quad (131)$$

and equation (129) does describe the symmetry setting $\mu = U/2$. For a detailed explanation of particle-hole symmetry, see [112], for instance.

Let us now formulate the Hubbard model within the path-integral formalism. The path-integral representation of the grand partition function is

$$\begin{aligned} Z = \int [\mathbf{d}\bar{\psi} \mathbf{d}\psi] \exp \left[- \int_0^\beta d\tau \left\{ \sum_{n \in \Lambda} \bar{\psi}(n, \tau) \left(\frac{\partial}{\partial \tau} - \mu \right) \psi(n, \tau) \right. \right. \\ \left. \left. - t \sum_n \sum_{\sigma=1}^d \left(\bar{\psi}(n + \hat{\sigma}, \tau) \psi(n, \tau) + \bar{\psi}(n, \tau) \psi(n + \hat{\sigma}, \tau) \right) \right. \right. \\ \left. \left. + \frac{U}{2} \left(\bar{\psi}(n, \tau) \psi(n, \tau) \right)^2 \right\} \right], \end{aligned} \quad (132)$$

where the two-component Grassmann variables ψ and $\bar{\psi}$ have been defined via

$$\psi(n, \tau) = \begin{bmatrix} \psi_\uparrow(n, \tau) \\ \psi_\downarrow(n, \tau) \end{bmatrix}, \quad \bar{\psi}(n, \tau) = [\bar{\psi}_\uparrow(n, \tau), \bar{\psi}_\downarrow(n, \tau)]. \quad (133)$$

The imaginary time is parameterized by τ and β is the inverse temperature. The Grassmann fields obey the anti-periodic boundary condition along the imaginary time direction. When the imaginary time direction is discretized by $\beta = N_\tau \epsilon$, we can identify

$$\begin{aligned} S = \sum_{n \in \Lambda'} \epsilon \left[\bar{\psi}(n) \frac{\psi(n + \hat{\tau}) - \psi(n)}{\epsilon} \right. \\ \left. - t \sum_{\sigma=1}^d \left(\bar{\psi}(n + \hat{\sigma}) \psi(n) + \bar{\psi}(n) \psi(n + \hat{\sigma}) \right) \right. \\ \left. + \frac{U}{2} \left(\bar{\psi}(n) \psi(n) \right)^2 - \mu \bar{\psi}(n) \psi(n) \right] \end{aligned} \quad (134)$$

as the action of the Hubbard model on the $(d+1)$ -dimensional anisotropic lattice Λ' . The unit vector along the imaginary time direction is denoted as $\hat{\tau}$ in equation (134). The Grassmann fields $\psi(n)$ and $\bar{\psi}(n)$ on Λ' are defined in the same manner with equation (133). With $\mu = U/2$, the system is at half-filling as before.

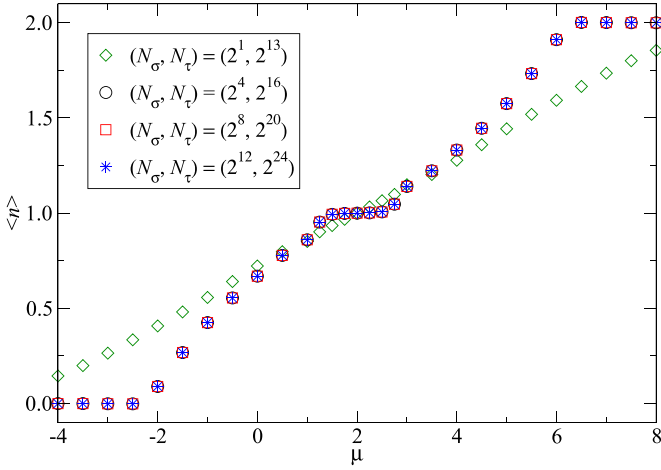


Figure 22. Adapted from [115]. Electron number density $\langle n \rangle$ at $(U, t) = (4, 1)$ with $D_{\text{HOTRG}} = 80$ and $\epsilon = 10^{-4}$. N_σ and N_τ denote the spatial and temporal lattice sizes, respectively. Reproduced from [115]. CC BY 4.0.

6.1. (1+1)-dimensional model

Akiyama and Kuramashi provided a benchmark study of the TRG method for the (1+1)-dimensional Hubbard model in [115]. They also derived the Grassmann tensor network representation of the path integral defined by equation (134) in general spatial dimensions. In equation (134), the time slice ϵ should be $\epsilon \ll 1$ to reproduce the grand partition function in equation (132). In other words, the contribution from the temporal hopping terms ($O(1)$) becomes much larger than that from the spatial ones ($O(\epsilon)$) in the original action. Consequently, the initial Grassmann tensor network becomes extremely anisotropic in the temporal direction. The authors in [115] applied the HOTRG algorithm along the temporal direction in advance before the spacetime coarse-graining took place to investigate the ground state.

Throughout their study, the number density $\langle n \rangle$ is computed as a function of the chemical potential μ on the three points $(U, t) = (4, 0), (0, 1), (4, 1)$. At finite U , the Mott plateau $\langle n \rangle = 1$ is reproduced as shown in figure 22. At $(U, t) = (4, 1)$, they provided the numerical fit of the number density via

$$\langle n \rangle = A + B|\mu - \mu_c(D_{\text{HOTRG}})|^\nu. \quad (135)$$

This fit gives the pseudo critical point $\mu_c(D_{\text{HOTRG}})$ and the exponent ν . With the bond dimension $D_{\text{HOTRG}} \in [60, 80]$, the resulting ν is consistent with the exact value $\nu = 0.5$. Extrapolating $\mu_c(D_{\text{HOTRG}})$ to the limit $D_{\text{HOTRG}} \rightarrow \infty$, they used the two fitting ansatz,

$$\mu_c(D_{\text{HOTRG}}) = \mu_c + aD_{\text{HOTRG}}^{-1}, \quad (136)$$

$$\mu_c(D_{\text{HOTRG}}) = \mu_c + bD_{\text{HOTRG}}^{-c}, \quad (137)$$

where the latter fit was employed to estimate uncertainty in the fitting assumption. These numerical fits are shown in figure 23. The extrapolation has given $\mu_c = 2.642(05)(13)$ which is in agreement with the exact value $\mu_c = 2.643 \dots$ based on the Bethe ansatz.

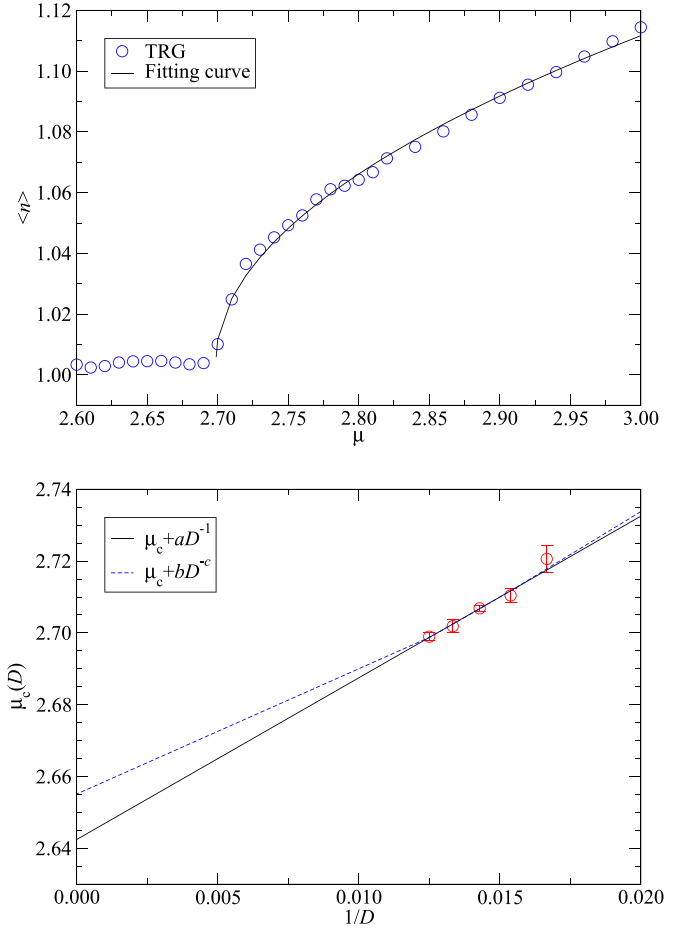


Figure 23. Adapted from [115]. (top) Electron number density $\langle n \rangle$ at $\beta = N_\tau \epsilon = 1677.7216$ with $\epsilon = 10^{-4}$ and $D_{\text{HOTRG}} = 80$. The fit ansatz in equation (135) results in $\mu_c(D_{\text{HOTRG}}) = 2.698(1)$ and $\nu = 0.51(2)$. (bottom) $\mu_c(D_{\text{HOTRG}})$ as a function of $1/D_{\text{HOTRG}}$. The solid line shows the fit result of equation (136) and the dotted curve gives the fit result of equation (137). Reproduced from [115]. CC BY 4.0.

6.2. (2+1)-dimensional model

Akiyama, Kuramashi, and Yamashita investigated the metal-insulator transition in the (2+1)-dimensional Hubbard model [116]. They used the ATRG algorithm to investigate the ground state and applied the same strategy with [115]; the spacetime coarse-graining after the imaginary-time evolution. As a validation of the numerical strategy, the number density at $(U, t) = (8, 0)$ is computed. The TRG result is consistent with the exact one and the Mott plateau is reproduced around $\mu = 4$ as expected. The number density is also computed at $(U, t) = (80, 1), (8, 1), (2, 1)$. With $(U, t) = (80, 1)$, the TRG gives the smooth number density as a function of the chemical potential and $\mu_c/U \neq 1$ in contrast to $(U, t) = (8, 0)$. These results imply that the TRG calculation captures the spatial hopping effects even with the large repulsion parameter U . At $(U, t) = (8, 1), (2, 1)$, the transition point μ_c is determined by the global fit using the quadratic function,

$$\langle n \rangle = 1 + a(\mu - \mu_c(D_{\text{ATRNG}})) + b(\mu - \mu_c(D_{\text{ATRNG}}))^2, \quad (138)$$

with

$$\mu_c(D_{\text{ATRG}}) = \mu_c + cD_{\text{ATRG}}^{-1}. \quad (139)$$

D_{ATRG} is varied as 56, 64, 72, 80. Their estimates are $\mu_c = 6.43(4)$ at $(U, t) = (8, 1)$ and $\mu_c = 1.30(6)$ at $(U, t) = (2, 1)$. Based on these estimates, the authors expect that $|\mu_c - U/2|$ vanishes only at $U = 0$, that is the metal–insulator transition could take place with any finite repulsion.

7. Conclusions

We have reviewed the two formulations to derive the Grassmann tensor network representations for fermionic path integrals. We have shown that both formulations can result in the same ordinary tensor (bosonic tensor or coefficient tensor) by properly ordering the auxiliary Grassmann variables. Exact contractions are defined as the integration of these auxiliary Grassmann variables. These formulations immediately allow us to extend any TRG algorithms for fermions. By introducing the Grassmann parity functions, we can immediately extend the algorithms, such as the Levin–Nave TRG and HOTRG, for fermionic path integrals under explicit correspondence with the original ones. In particular, the geometric representation and the connectivities remain identical.

These Grassmann TRG algorithms have been applied to various lattice theories including the relativistic models not only in two but also in four dimensions and the Hubbard model at finite density. Some of these models include examples where the Monte Carlo method is extremely difficult to apply due to the sign problem.

We have also reviewed several TNR algorithms, the bond-weighting method, and multilayered Grassmann tensor networks for N_f -flavor fermions. Although some of these improved TRG methods were originally proposed for spin systems, recent numerical calculations have shown that these methods are also efficient for fermionic systems. Research on such improved algorithms has continued to progress in recent years; a new algorithm of loop-TNR [181], combinations of the Monte Carlo method and TRG [182–184], and application of the machine-learning techniques to the TRG [185–187]. The extension of these novel improved methods to fermionic systems is considered important in examining whether these methods are also valid for more general physical systems including fermions.

Data availability statement

Elements of the data that support the findings of this study are openly available at the following URL/DOI: <https://github.com/akiyama-es/Grassmann-BTRG>.

Acknowledgment

We thank M Hite and other members of the QuLAT Collaboration for valuable discussions. Y M was supported in part by the U.S. Department of Energy (DOE) under

Award Number DE-SC0019139 and DE-SC0010113. This research used resources of the Syracuse University HTC Campus Grid and NSF award ACI-1341006 and the National Energy Research Scientific Computing Center (NERSC), a U.S. Department of Energy Office of Science User Facility located at Lawrence Berkeley National Laboratory, operated under Contract No. DE-AC02-05CH11231 using NERSC awards HEP-ERCAP0020659 and HEP-ERCAP0023235. S A acknowledges the support from the Endowed Project for Quantum Software Research and Education, the University of Tokyo (<https://qsw.phys.s.u-tokyo.ac.jp/>) and JSPS KAKENHI Grant Number JP23K13096.

ORCID iDs

Shinichiro Akiyama  <https://orcid.org/0000-0003-1415-5620>

Yannick Meurice  <https://orcid.org/0000-0002-0995-9694>

Ryo Sakai  <https://orcid.org/0000-0002-0064-2041>

References

- [1] Kadanoff L P 1966 *Phys. Phys. Fiz.* **2** 263–72
- [2] Wilson K G and Kogut J B 1974 *Phys. Rep.* **12** 75–199
- [3] Wilson K G 1975 *Rev. Mod. Phys.* **47** 773
- [4] Cardy J 1996 *Scaling and Renormalization in Statistical Physics (Cambridge Lecture Notes in Physics)* (Cambridge University Press)
- [5] Migdal A A 1975 *Sov. Phys. JETP* **42** 743
- [6] Kadanoff L P 1976 *Ann. Phys.* **100** 359–94
- [7] Dyson F J 1969 *Commun. Math. Phys.* **12** 91–107
- [8] Baker G A 1972 *Phys. Rev. B* **5** 2622–33
- [9] Meurice Y 2007 *J. Phys. A: Math. Theor.* **40** R39
- [10] Berges J, Tetradis N and Wetterich C 2002 *Phys. Rep.* **363** 223–386
- [11] Bervillier C, Juttner A and Litim D F 2007 *Nucl. Phys. B* **783** 213–26
- [12] Bervillier C 2013 *Nucl. Phys. B* **876** 587–604
- [13] White S R 1992 *Phys. Rev. Lett.* **69** 2863–6
- [14] Schollwöck U 2005 *Rev. Mod. Phys.* **77** 259–315
- [15] Schollwöck U 2011 *Ann. Phys., NY* **326** 96–192
- [16] Vidal G 2007 *Phys. Rev. Lett.* **99** 220405
- [17] Cirac J I and Verstraete F 2009 *J. Phys. A: Math. Theor.* **42** 504004
- [18] Schollwöck U 2011 *Phil. Trans. R. Soc. A* **369** 2643–61
- [19] Orús R 2014 *Ann. Phys.* **349** 117–58
- [20] Evenbly G and Vidal G 2015 *Phys. Rev. Lett.* **115** 180405
- [21] Silvi P, Tschirsich F, Gerster M, Jünemann J, Jaschke D, Rizzi M and Montangero S 2019 *SciPost Phys. Lect. Notes* **8** 1
- [22] Haegeman J and Verstraete F 2017 *Ann. Rev. Condens. Matter Phys.* **8** 355–406
- [23] Montangero S 2018 *Introduction to Tensor Network Methods: Numerical Simulations of low-Dimensional Many-Body Quantum Systems* (Springer)
- [24] Ran S J, Tirrito E, Peng C, Chen X, Tagliacozzo L, Su G and Lewenstein M 2020 *Tensor Network Contractions: Methods and Applications to Quantum Many-Body Systems* (Springer)
- [25] Cirac J I, Perez-Garcia D, Schuch N and Verstraete F 2021 *Rev. Mod. Phys.* **93** 045003
- [26] Nishino T and Okunishi K 1996 *J. Phys. Soc. Japan* **65** 891
- [27] Levin M and Nave C P 2007 *Phys. Rev. Lett.* **99** 120601
- [28] Gu Z C and Wen X G 2009 *Phys. Rev. B* **80** 155131

- [29] Xie Z Y, Jiang H C, Chen Q N, Weng Z Y and Xiang T 2009 *Phys. Rev. Lett.* **103** 160601
- [30] Gu Z C, Verstraete F and Wen X G 2010 arXiv:1004.2563
- [31] Gu Z C 2013 *Phys. Rev. B* **88** 115139
- [32] Xie Z Y, Chen J, Qin M P, Zhu J W, Yang L P and Xiang T 2012 *Phys. Rev. B* **86** 045139
- [33] Efrati E, Wang Z, Kolan A and Kadanoff L P 2014 *Rev. Mod. Phys.* **86** 647–67
- [34] Meurice Y 2013 *Phys. Rev. B* **87** 064422
- [35] Lyu X, Xu R G and Kawashima N 2021 *Phys. Rev. Res.* **3** 023048
- [36] Ueda A and Oshikawa M 2021 *Phys. Rev. B* **104** 165132
- [37] Ueda A and Oshikawa M 2022 *Phys. Rev. E* **106** 014104
- [38] Ueda A and Oshikawa M 2023 *Phys. Rev. B* **108** 024413
- [39] Huang C Y, Chan S H, Kao Y J and Chen P 2023 *Phys. Rev. B* **107** 205123
- [40] Guo W and Wei T C 2023 arXiv:2305.09899
- [41] Ueda A and Yamazaki M 2023 arXiv:2307.02523
- [42] Meurice Y, Perry R and Tsai S W 2011 *Phil. Trans. R. Soc. A* **369** 2602–11
- [43] Byrnes T, Sriganesh P, Bursill R J and Hamer C J 2002 *Phys. Rev. D* **66** 013002
- [44] Bañuls M, Cichy K, Jansen K and Cirac J 2013 *J. High Energy Phys.* **JHEP11(2013)158**
- [45] Buyens B, Haegeman J, Van Acoleyen K, Verschelde H and Verstraete F 2014 *Phys. Rev. Lett.* **113** 091601
- [46] Buyens B, Haegeman J, Verschelde H, Verstraete F and Van Acoleyen K 2016 *Phys. Rev. X* **6** 041040
- [47] Funcke L, Jansen K and Kühn S 2020 *Phys. Rev. D* **101** 054507
- [48] Dempsey R, Klebanov I R, Pufu S S and Zan B 2022 *Phys. Rev. Res.* **4** 043133
- [49] Okuda T 2023 *Phys. Rev. D* **107** 054506
- [50] Honda M, Itou E and Tanizaki Y 2022 *J. High Energy Phys.* **JHEP11(2022)141**
- [51] Itou E, Matsumoto A and Tanizaki Y 2023 *J. High Energy Phys.* **JHEP11(2023)231**
- [52] Kühn S, Cirac J I and Bañuls M C 2015 *J. High Energy Phys.* **JHEP07(2015)130**
- [53] Bañuls M C, Cichy K, Cirac J I, Jansen K and Kühn S 2017 *Phys. Rev. X* **7** 041046
- [54] Hayata T, Hidaka Y and Nishimura K 2023 arXiv:2311.11643
- [55] Liu H, Bhattacharya T, Chandrasekharan S and Gupta R 2023 arXiv:2312.17734
- [56] Bruckmann F, Jansen K and Kühn S 2019 *Phys. Rev. D* **99** 074501
- [57] Tagliacozzo L, Celi A and Lewenstein M 2014 *Phys. Rev. X* **4** 041024
- [58] Silvi P, Rico E, Dalmonte M, Tschirsich F and Montangero S 2017 *Quantum* **1** 9
- [59] Silvi P, Sauer Y, Tschirsich F and Montangero S 2019 *Phys. Rev. D* **100** 074512
- [60] Rico E, Pichler T, Dalmonte M, Zoller P and Montangero S 2014 *Phys. Rev. Lett.* **112** 201601
- [61] Pichler T, Dalmonte M, Rico E, Zoller P and Montangero S 2016 *Phys. Rev. X* **6** 011023
- [62] Zohar E and Burrello M 2015 *Phys. Rev. D* **91** 054506
- [63] Bañuls M C, Cichy K, Cirac J I, Jansen K and Kühn S 2018 *PoS (LATTICE 2018)* p 022
- [64] Bañuls M C and Cichy K 2020 *Rep. Prog. Phys.* **83** 024401
- [65] Balian R, Drouffe J M and Itzykson C 1975 *Phys. Rev. D* **11** 2104–19
- [66] Itzykson C and Drouffe J 1992 *Statistical Field Theory (Cambridge Monographs on Mathematical Physics)* (Cambridge University Press)
- [67] Liu Y, Meurice Y, Qin M P, Unmuth-Yockey J, Xiang T, Xie Z Y, Yu J F and Zou H 2013 *Phys. Rev. D* **88** 056005
- [68] Yu J F, Xie Z Y, Meurice Y, Liu Y, Denbleyker A, Zou H, Qin M P, Chen J and Xiang T 2014 *Phys. Rev. E* **89** 013308
- [69] Denbleyker A, Liu Y, Meurice Y, Qin M P, Xiang T, Xie Z Y, Yu J F and Zou H 2014 *Phys. Rev. D* **89** 016008
- [70] Zou H, Liu Y, Lai C Y, Unmuth-Yockey J, Bazavov A, Xie Z Y, Xiang T, Chandrasekharan S, Tsai S W and Meurice Y 2014 *Phys. Rev. A* **90** 063603
- [71] Akiyama S, Jha R G and Unmuth-Yockey J 2023 arXiv:2312.11649
- [72] Meurice Y 2019 *Phys. Rev. D* **100** 014506
- [73] Meurice Y 2020 *Phys. Rev. D* **102** 014506
- [74] Meurice Y, Sakai R and Unmuth-Yockey J 2022 *Rev. Mod. Phys.* **94** 025005
- [75] Barthel T, Pineda C and Eisert J 2009 *Phys. Rev. A* **80** 042333
- [76] Corboz P, Evenbly G, Verstraete F and Vidal G 2010 *Phys. Rev. A* **81** 010303
- [77] Nishino T, Hieida Y, Okunishi K, Maeshima N, Akutsu Y and Gendiar A 2001 *Prog. Theor. Phys.* **105** 409–17
- [78] Gendiar A, Maeshima N and Nishino T 2003 *Prog. Theor. Phys.* **110** 691–9
- [79] Verstraete F and Cirac J I 2004 arXiv:cond-mat/0407066
- [80] Kraus C V, Schuch N, Verstraete F and Cirac J I 2010 *Phys. Rev. A* **81** 052338
- [81] Zohar E and Burrello M 2016 *New J. Phys.* **18** 043008
- [82] Zohar E and Cirac J I 2018 *Phys. Rev. D* **97** 034510
- [83] Zapp K and Orús R 2017 *Phys. Rev. D* **95** 114508
- [84] Felser T, Silvi P, Collura M and Montangero S 2020 *Phys. Rev. X* **10** 041040
- [85] Magnifico G, Felser T, Silvi P and Montangero S 2021 *Nat. Commun.* **12** 3600
- [86] Felser T, Notarnicola S and Montangero S 2021 *Phys. Rev. Lett.* **126** 170603
- [87] Emonts P, Bañuls M C, Cirac I and Zohar E 2020 *Phys. Rev. D* **102** 074501
- [88] Robaina D, Bañuls M C and Cirac J I 2021 *Phys. Rev. Lett.* **126** 050401
- [89] Emonts P, Kelman A, Borla U, Moroz S, Gazit S and Zohar E 2023 *Phys. Rev. D* **107** 014505
- [90] Bender J, Emonts P and Cirac J I 2023 *Phys. Rev. Res.* **5** 043128
- [91] Cataldi G, Magnifico G, Silvi P and Montangero S 2023 arXiv:2307.09396
- [92] Emonts P and Zohar E 2023 *Phys. Rev. D* **108** 014514
- [93] Emonts P and Zohar E 2020 *SciPost Phys. Lect. Notes* **12** 1
- [94] Troyer M and Wiese U J 2005 *Phys. Rev. Lett.* **94** 170201
- [95] Prokof'ev N and Svistunov B 2007 *Phys. Rev. Lett.* **99** 250201
- [96] Chandrasekharan S 2010 *Phys. Rev. D* **82** 025007
- [97] Dornheim T 2019 *Phys. Rev. E* **100** 023307
- [98] Dornheim T 2021 *J. Phys. A: Math. Theor.* **54** 335001
- [99] Vanhecke B, Haegeman J, Van Acoleyen K, Vanderstraeten L and Verstraete F 2019 *Phys. Rev. Lett.* **123** 250604
- [100] Yang S, Gu Z C and Wen X G 2017 *Phys. Rev. Lett.* **118** 110504
- [101] Hauru M, Delcamp C and Mizera S 2018 *Phys. Rev. B* **97** 045111
- [102] Takeda S and Yoshimura Y 2015 *Prog. Theor. Exp. Phys.* **2015** 043B01
- [103] Akiyama S 2023 *Phys. Rev. D* **108** 034514
- [104] Bloch J and Lohmayer R 2023 *Nucl. Phys. B* **986** 116032
- [105] Akiyama S, Kuramashi Y, Yamashita T and Yoshimura Y 2021 *J. High Energy Phys.* **JHEP01(2021)121**
- [106] Kadoh D, Kuramashi Y, Nakamura Y, Sakai R, Takeda S and Yoshimura Y 2018 *J. High Energy Phys.* **JHEP03(2018)141**
- [107] Asaduzzaman M, Catterall S, Meurice Y, Sakai R and Toga G C 2023 arXiv:2312.16167
- [108] Dagotto E 1994 *Rev. Mod. Phys.* **66** 763–840

- [109] Georges A, Kotliar G, Krauth W and Rozenberg M J 1996 *Rev. Mod. Phys.* **68** 13–125
- [110] Tasaki H 1998 *J. Phys.: Condens. Matter* **10** 4353
- [111] LeBlanc J P F *et al* (Simons Collaboration on the Many-Electron Problem) 2015 *Phys. Rev. X* **5** 041041
- [112] Arovas D P, Berg E, Kivelson S A and Raghu S 2022 *Annu. Rev. Condens. Matter Phys.* **13** 239–74
- [113] Qin M, Schäfer T, Andergassen S, Corboz P and Gull E 2022 *Annu. Rev. Condens. Matter Phys.* **13** 275–302
- [114] Ostmeyer J 2023 *PoS (LATTICE 2022)* p 230
- [115] Akiyama S and Kuramashi Y 2021 *Phys. Rev. D* **104** 014504
- [116] Akiyama S, Kuramashi Y and Yamashita T 2022 *Prog. Theor. Exp. Phys.* **2022** 023101
- [117] Shimizu Y and Kuramashi Y 2014 *Phys. Rev. D* **90** 014508
- [118] Shimizu Y and Kuramashi Y 2014 *Phys. Rev. D* **90** 074503
- [119] Shimizu Y and Kuramashi Y 2018 *Phys. Rev. D* **97** 034502
- [120] Asaduzzaman M, Catterall S, Meurice Y, Sakai R and Toga G C 2023 *J. High Energy Phys.* **JHEP01(2023)024**
- [121] Sakai R, Takeda S and Yoshimura Y 2017 *Prog. Theor. Exp. Phys.* **2017** 063B07
- [122] Yoshimura Y, Kuramashi Y, Nakamura Y, Takeda S and Sakai R 2018 *Phys. Rev. D* **97** 054511
- [123] Meurice Y 2018 *PoS (LATTICE 2018)* p 231
- [124] Bao C 2019 Loop optimization of tensor network renormalization: algorithms and applications *PhD Thesis* U. Waterloo (main)
- [125] Akiyama S and Kadoh D 2021 *J. High Energy Phys.* **JHEP10(2021)188**
- [126] Akiyama S 2022 *J. High Energy Phys.* **JHEP11(2022)030**
- [127] Yosprakob A, Nishimura J and Okunishi K 2023 *J. High Energy Phys.* **JHEP11(2023)187**
- [128] Akiyama S 2023 arXiv:2311.17691
- [129] Yosprakob A 2023 arXiv:2309.07557
- [130] Okunishi K, Nishino T and Ueda H 2022 *J. Phys. Soc. Japan* **91** 062001
- [131] Akiyama S, Kuramashi Y and Yoshimura Y 2022 *PoS (LATTICE 2021)* p 530
- [132] Kadoh D 2022 *PoS (LATTICE 2021)* p 633
- [133] Kuramashi Y and Yoshimura Y 2020 *J. High Energy Phys.* **JHEP04(2020)089**
- [134] Fukuma M, Kadoh D and Matsumoto N 2021 *Prog. Theor. Exp. Phys.* **2021** 123B03
- [135] Hirasawa M, Matsumoto A, Nishimura J and Yosprakob A 2021 *J. High Energy Phys.* **JHEP12(2021)011**
- [136] Dittrich B, Mizera S and Steinhaus S 2016 *New J. Phys.* **18** 053009
- [137] Kuwahara T and Tsuchiya A 2022 *Prog. Theor. Exp. Phys.* **2022** 093B02
- [138] Luo X and Kuramashi Y 2023 *Phys. Rev. D* **107** 094509
- [139] Akiyama S and Kuramashi Y 2022 *J. High Energy Phys.* **JHEP05(2022)102**
- [140] Morita S, Igarashi R, Zhao H H and Kawashima N 2018 *Phys. Rev. E* **97** 033310
- [141] Wang L and Verstraete F 2011 arXiv:1110.4362
- [142] Corboz P, Rice T M and Troyer M 2014 *Phys. Rev. Lett.* **113** 046402
- [143] Iino S, Morita S and Kawashima N 2019 *Phys. Rev. B* **100** 035449
- [144] Zhao H H, Xie Z Y, Chen Q N, Wei Z C, Cai J W and Xiang T 2010 *Phys. Rev. B* **81** 174411
- [145] Evenbly G 2017 *Phys. Rev. B* **95** 045117
- [146] Orús R and Vidal G 2008 *Phys. Rev. B* **78** 155117
- [147] Wolff U 2020 *Nucl. Phys. B* **955** 115061
- [148] Gausterer H and Lang C B 1995 *Nucl. Phys. B* **455** 785–95
- [149] Butt N, Catterall S, Meurice Y, Sakai R and Unmuth-Yockey J 2020 *Phys. Rev. D* **101** 094509
- [150] Gattringer C, Kloiber T and Sazonov V 2015 *Nucl. Phys. B* **897** 732–48
- [151] Göschl D, Gattringer C, Lehmann A and Weis C 2017 *Nucl. Phys. B* **924** 63–85
- [152] Eckart C and Young G 1936 *Psychometrika* **1** 211–8
- [153] Adachi D, Okubo T and Todo S 2022 *Phys. Rev. B* **105** L060402
- [154] Nakayama K, Funcke L, Jansen K, Kao Y J and Kühn S 2022 *Phys. Rev. D* **105** 054507
- [155] Adachi D 2020 High-accuracy tensor renormalization group algorithms and their applications *PhD Thesis* The University of Tokyo
- [156] Östlund S and Rommer S 1995 *Phys. Rev. Lett.* **75** 3537–40
- [157] Dukelsky J, Martín-Delgado M A, Nishino T and Sierra G 1998 *Europhys. Lett.* **43** 457–62
- [158] Vidal G 2003 *Phys. Rev. Lett.* **91** 147902
- [159] Rossi P and Wolff U 1984 *Nucl. Phys. B* **248** 105–22
- [160] Karsch F and Mütter K H 1989 *Nucl. Phys. B* **313** 541–59
- [161] Fromm M 2010 Lattice QCD at strong coupling *PhD Thesis* Zurich, ETH
- [162] Milde P, Bloch J and Lohmayer R 2022 *PoS (LATTICE 2021)* p 462
- [163] Bloch J, Lohmayer R, Meister M and Nunhofer M 2023 *Nucl. Phys. B* **987** 116107
- [164] Bloch J, Jha R G, Lohmayer R and Meister M 2021 *Phys. Rev. D* **104** 094517
- [165] Akiyama S 2022 Tensor renormalization group approach to higher-dimensional lattice field theories *PhD Thesis* University of Tsukuba
- [166] Adachi D, Okubo T and Todo S 2020 *Phys. Rev. B* **102** 054432
- [167] Oba H 2020 *Prog. Theor. Exp. Phys.* **2020** 013B02
- [168] Kadoh D and Nakayama K 2019 arXiv:1912.02414
- [169] Nakayama K 2023 arXiv:2307.14191
- [170] Bender I, Hashimoto T, Karsch F, Linke V, Nakamura A, Plewnia M, Stamatescu I O and Wetzel W 1992 *Nucl. Phys. B* **26** 323–5
- [171] Buballa M 2005 *Phys. Rep.* **407** 205–376
- [172] Aoki K I, Kumamoto S I and Yamada M 2018 *Nucl. Phys. B* **931** 105–31
- [173] Witten E 1982 *Nucl. Phys. B* **202** 253
- [174] Catterall S, Kaplan D B and Unsal M 2009 *Phys. Rep.* **484** 71–130
- [175] De Lathauwer L, De Moor B and Vandewalle J 2000 *SIAM J. Matrix Anal. Appl.* **21** 1253–78
- [176] De Lathauwer L, De Moor B and Vandewalle J 2000 *SIAM J. Matrix Anal. Appl.* **21** 1324–42
- [177] van den Doel C and Smit J 1983 *Nucl. Phys. B* **228** 122–44
- [178] Lieb E H and Wu F Y 1968 *Phys. Rev. Lett.* **20** 1445–8
- [179] Metzner W and Vollhardt D 1989 *Phys. Rev. Lett.* **62** 324–7
- [180] Müller-Hartmann E 1989 *Z. Phys. B: Condens. Matter* **74** 507–12
- [181] Homma K and Kawashima N 2023 arXiv:2306.17479
- [182] Ferris A J 2015 arXiv:1507.00767
- [183] Huggins W, Freeman C D, Stoudenmire M, Tubman N M and Whaley K B 2017 arXiv:1710.03757
- [184] Arai E, Ohki H, Takeda S and Tomii M 2023 *Phys. Rev. D* **107** 114515
- [185] Liao H J, Liu J G, Wang L and Xiang T 2019 *Phys. Rev. X* **9** 031041
- [186] Chen B B, Gao Y, Guo Y B, Liu Y, Zhao H H, Liao H J, Wang L, Xiang T, Li W and Xie Z Y 2020 *Phys. Rev. B* **101** 220409
- [187] Jha R G and Samlodia A 2024 *Comput. Phys. Commun.* **294** 108941