







Automatic differentiation for second renormalization of tensor networksBin-Bin Chen ^{1,2}, Yuan Gao,¹ Yi-Bin Guo ^{3,4}, Yuzhi Liu ⁵, Hui-Hai Zhao ⁶, Hai-Jun Liao,^{3,7}
Lei Wang,^{3,7} Tao Xiang,^{3,4,8} Wei Li ^{1,*} and Z. Y. Xie ^{9,†}¹*School of Physics, Key Laboratory of Micro-Nano Measurement-Manipulation and Physics (Ministry of Education),
Beihang University, Beijing 100191, China*²*Physics Department, Arnold Sommerfeld Center for Theoretical Physics, and Center for NanoScience,
Ludwig-Maximilians-Universität, Theresienstrasse 37, 80333 Munich, Germany*³*Institute of Physics, Chinese Academy of Sciences, P. O. Box 603, Beijing 100190, China*⁴*School of Physics, University of Chinese Academy of Sciences, Beijing 100049, China*⁵*Department of Physics, Indiana University, Bloomington, Indiana 47405, USA*⁶*Alibaba Quantum Laboratory, Alibaba Group, Beijing, China*⁷*Songshan Lake Materials Laboratory, Dongguan, Guangdong 523808, China*⁸*Kavli Institute for Theoretical Sciences, University of Chinese Academy of Sciences, Beijing 100190, China*⁹*Department of Physics, Renmin University of China, Beijing 100872, China*(Received 25 December 2019; revised manuscript received 2 June 2020; accepted 4 June 2020;
published 23 June 2020)

Tensor renormalization group (TRG) constitutes an important methodology for accurate simulations of strongly correlated lattice models. Facilitated by the automatic differentiation technique widely used in deep learning, we propose a uniform framework of differentiable TRG (∂ TRG) that can be applied to improve various TRG methods, in an automatic fashion. ∂ TRG systematically extends the essential concept of second renormalization [Phys. Rev. Lett. **103**, 160601 (2009)] where the tensor environment is computed recursively in the backward iteration. Given the forward TRG process, ∂ TRG automatically finds the gradient of local tensors through backpropagation, with which one can deeply “train” the tensor networks. We benchmark ∂ TRG in solving the square-lattice Ising model, and we demonstrate its power by simulating one- and two-dimensional quantum systems at finite temperature. The global optimization as well as GPU acceleration renders ∂ TRG a highly efficient and accurate many-body computation approach.

DOI: [10.1103/PhysRevB.101.220409](https://doi.org/10.1103/PhysRevB.101.220409)

Introduction. In the investigation of strongly correlated quantum states and materials, tensor renormalization group (TRG) constitutes a thriving field that has been playing an increasingly important role recently. The diverse family of TRG approaches includes the coarse-graining TRG [1,2], higher-order TRG (HOTRG) [3], and tensor network renormalization [4–7]. They have been put forward to evaluate classical statistical systems as well as expectation values out of two-dimensional (2D) tensor network states [8]. There are also TRG methods developed to simulate d -dimensional quantum lattice models at finite temperature [9–17], whose Euclidean path integral constitutes a $(d + 1)$ -dimensional worldsheet.

In the course of TRG process, the environment of local tensors should be taken into account to conduct a precise truncation through, e.g., isometric renormalization transformations in the tensor bases. This can be traced back to the renowned density matrix renormalization group [18], where the effects of the environment are reflected in the reduced density matrix of the “system” subblock. For generic tensor networks, a second renormalization group (SRG) has been proposed to improve the process of tensor renormalization [3,19–21]. In

SRG, the environment of local tensors is computed recursively between different scales of a hierarchical network, with which a global optimization is feasible.

Recently, the profound interplay between deep learning and tensor network algorithms has attracted a great deal of interest [22–30]. Among other things, the differentiable programming is of particular interest for tensor networks. Given the computational graph generated in the forward process, the gradient of corresponding variables can be calculated through the chain rule of derivatives in the backpropagation, with which the neural network can be deeply trained. The automatic calculation of gradients can be obtained within machine precision, and with the same computational complexity as in the forward process. Very recently, this idea of differentiable programming that exploits a gradient-based optimization has been introduced to optimize tensor networks [31].

In this work, we regard the renormalization transformation as input parameters of the TRG program, and we point out that the SRG backward iteration corresponds with the backpropagation algorithm in differentiable programming. Inspired by this substantial connection, we turn the idea of SRG into a generalized versatile framework, i.e., differentiable TRG (∂ TRG).

In ∂ TRG, the forward TRG process is made fully differentiable, and the renormalization transformations are optimized

*w.li@buaa.edu.cn

†qingtaoxie@ruc.edu.cn

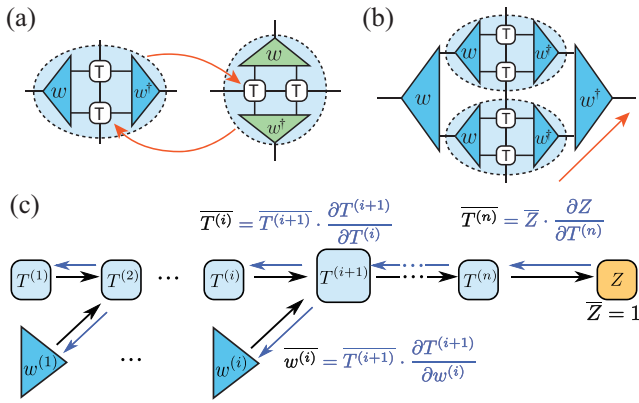


FIG. 1. Part (a) shows a TRG step where the scale transformations w are introduced along both directions, while only vertical renormalization is involved in (b), which eventually compresses the tensor network into a 1D structure that can then be contracted exactly. Part (c) plots the computational graph of the forward TRG process as well as the backpropagation.

globally and automatically through backpropagation. We apply ∂ TRG to simulate thermal equilibrium states at finite temperature, and we achieve significantly improved accuracy over previous methods [9,16]. The efficiency is demonstrated by implementing ∂ TRG with PyTorch [32,33], which facilitates the GPU computing and shows a high performance of about 40 times acceleration over a single CPU core.

Correspondence between SRG and backpropagation. Backpropagation is a widely used method for training deep neural networks [34–37], where the gradients of parameters can be computed through a reverse-mode automatic differentiation [38]. On the other hand, in tensor-network algorithms, SRG plays a very similar role to backpropagation. To be specific, as shown in Figs. 1(a) and 1(c), a hierarchical tensor network can be constructed by piling up a series of isometric RG transformations $\{w^{(i)}\}$, with $i = 1, 2, \dots, n$ for each layer. In a well-designed TRG program, the input tensors $\{w^{(i)}\}$ are successively applied to the tensors T , and the output can be a tensor trace in general, e.g., a partition function of a statistical system.

SRG further optimizes the renormalization transformations $\{w^{(i)}\}$ in a backward iteration by making use of the environment. An adjoint tensor of $w^{(i)}$ at scale i is defined as the gradient $\overline{w^{(i)}} (\equiv \frac{\partial Z}{\partial w^{(i)}})$, which can be related to the environment through

$$E_w^{(i)} = \frac{1}{N_i} \overline{w^{(i)}}, \quad (1)$$

where N_i denotes the times $w^{(i)}$ appearing in the network, and Z is the tensor trace to be maximized. It directly follows from Eq. (1) that the recursive relations used in the backward iteration to determine $\{E^{(i)}\}$ in SRG can be recast into the derivative chain rule form, as depicted in Fig. 1(c). Recall that the multiplications there with Jacobians $\partial T^{(i+1)}/\partial T^{(i)}$ and $\partial T^{(i+1)}/\partial w^{(i)}$, etc., are conducted implicitly. They constitute sequences of tensor contractions exactly equivalent to the recursive tensor contractions in SRG [38].

Differentiable tensor renormalization group. Being aware of the intimate relation between backpropagation and SRG, we now extend the latter to a more general and flexible framework, ∂ TRG, with the help of well-developed automatic differentiation packages [39], e.g., AUTOGRAD [40] and PYTORCH [32,33]. With these facilities, ∂ TRG can record all operations performed on the input variables (tensors), and compute the derivatives [e.g. Eq. (1)] automatically in the backward iterations with which the parameters $\{w^{(i)}\}$ can be optimized. Not limited within the original proposals [3,19], the idea of SRG can be applied to various TRG schemes through the framework of *differentiable programming*. Below we consider two different ∂ TRG schemes following the HOTRG [3] and exponential TRG (XTRG) [16], as shown in Figs. 1(a) and 1(b), respectively.

Once the environment $E_w^{(i)}$ is obtained, one can optimize $w^{(i)}$ with resorting to, e.g., the standard quasi-Newton optimization method [38], or quasioptimal schemes through tensor decompositions of $E_w^{(i)}$ [3,19]. Recall that a tensor decomposition scheme that keeps $w^{(i)}$ isometric has been developed in the context of the multiscale entanglement renormalization ansatz (MERA) algorithm [41,42], which is mainly adopted in the simulations below. MERA update involves a singular value decomposition (SVD) $E_w = USV^\dagger$ and a replacement $w = UV^\dagger$, which maximizes the cost function $Z = \text{Tr}(E_w w)$, with $O(D^4)$ time complexity. Here D is the geometric bond dimension of a tensor.

Due to the intrinsic nonlinearity (in w) in the optimization problem, n_i inner iterations are introduced in a single step of MERA update ($n_i = 5-10$ in practice). Moreover, thanks to the convenient access to $E_w^{(i)}$, in ∂ TRG we can deeply optimize the tensor network via sweep optimizations. In practice, we scan from inner to outer layers n_s times until the results converge, thus assuring a highly accurate global update of $\{w^{(i)}\}$ tensors.

∂ TRG of the 2D Ising model. As a first demonstration, we apply ∂ TRG, with two specific implementations in Figs. 1(a) and 1(b) to solve the classical Ising model on the square lattice. Following the standard procedure, we can write down a square tensor-network representation consisting of rank-4 tensors T , whose TRG contraction results in the partition function Z [38].

In Fig. 1(a), after n steps of renormalizations, we obtain a single tensor representing the whole system of $2^n \times 2^n$ sites (in practice, $n = 25$ guarantees the thermodynamic limit), whose self-contraction leads to the partition function Z . On the other hand, after n steps of renormalization, one arrives at an effective one-dimensional (1D) system, whose complete contraction also leads to an accurate measure of the partition function.

In Fig. 2, we show the accuracies of ∂ TRG implementations, together with the HOTRG and HOSRG data for comparisons. Due to the sweep update, ∂ TRG leads to errors clearly smaller than those of HOTRG, while achieving, as expected, the same accuracies as HOSRG [43].

The two schemes of ∂ TRG in Figs. 1(a) and 1(b) have different computational costs. The latter is considerably less resource-demanding, i.e., $O(D^4)$ in computational time, while it is $O(D^7)$ in Fig. 1(a). The memory costs are also dramatically different, i.e., $O(D^5)$ for Fig. 1(a) and $O(D^3)$ for 1(b).

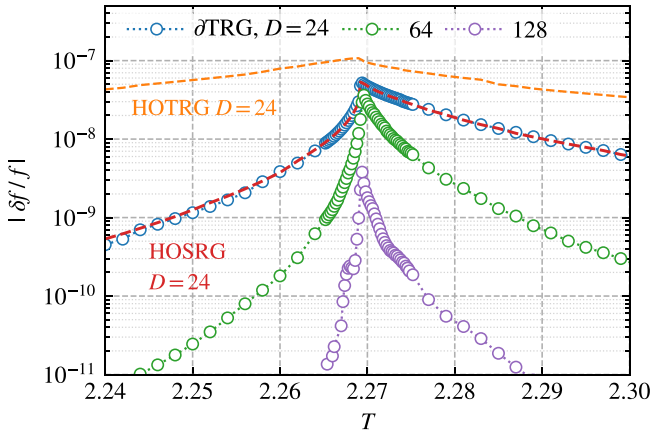


FIG. 2. The relative errors of free energy $|\delta f/f|$ in the vicinity of critical temperature T_c , obtained from HOTRG, HOSRG, and ∂ TRG calculations. We perform $n_i = 5-10$ iterations within each MERA update, and the total number of sweeps n_s ranges from a few iterations to a few hundred, depending on the specific temperature and scheme, until the relative errors reach a convergence criterion of $\epsilon = 10^{-4}$. The $D = 24$ ∂ TRG calculations follow the scheme in Fig. 1(a), while the $D = 64$ and 128 cases follow the scheme in Fig. 1(b).

Therefore, we can push the ∂ TRG simulations in Fig. 1(b) with bond states up to $D = 128$, reaching much higher precision as shown in Fig. 2. From the comparison shown in Fig. 2, as well as other considerations, we chose the ∂ TRG scheme in Fig. 1(b) to simulate quantum models as presented below.

Infinite quantum XY chain. Now we employ ∂ TRG to simulate the exactly solvable quantum XY chain

$$H_{XY} = \sum_{(i,j)} S_i^x S_j^x + S_i^y S_j^y. \quad (2)$$

We start by preparing the density matrix $\rho(\tau)$ through (second-order) Trotter-Suzuki decomposition [38]. A very small imaginary-time step τ is chosen to ensure that Trotter errors are negligible. Given the matrix product operator (MPO) representation of $\rho(\tau)$, we proceed to cool down the system exponentially fast, with the ∂ TRG algorithm shown in Fig. 1(b). The results are shown in Fig. 3(a), where the relative error $|\delta f/f|$ curves rise up from very small values at high temperature and increase monotonically as T decreases.

In Fig. 3(a), benchmarking with the analytical solution [38,44,45], we compare the relative errors $|\delta f/f|$ between ∂ TRG and LTRG, where the latter follows a cooling procedure linear in β ($\equiv 1/T$) [9]. It is observed that ∂ TRG with depth $n_d = 1$ (i.e., optimizing exclusively the current layer in the course of cooling) already outperforms LTRG in both efficiency and accuracy. By sweeping into n_d (up to four) layers, the accuracy is found to improve continuously in the relatively high to intermediate temperature regime due to better optimization. At low temperature, on the other hand, the enhancement of accuracy is marginal due to the limited expressibility of the tensor network with a given bond dimension $D = 32$. Therefore, we show also in Fig. 3(a) the results of larger bond dimensions (up to 512), with a fixed depth $n_d = 4$. There we observe that $|\delta f/f|$ decreases monotonically and

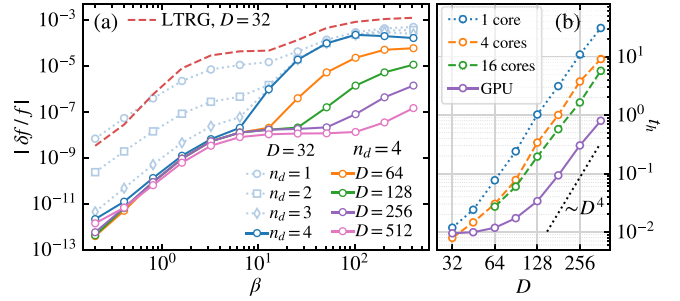


FIG. 3. (a) Comparisons of relative errors of free energy between linearized TRG (LTRG) and ∂ TRG. In the initial $\rho(\tau)$ of ∂ TRG, $\tau \simeq 5 \times 10^{-5}$, which is also used as the Trotter slice in the LTRG calculations. The results are shown with optimization depth $n_d \leq 4$, $n_i = 10$ in a single MERA update, and overall sweep iterations $n_s = 3$. (b) Comparisons of the computational wall time of ∂ TRG on GPU and CPU with up to 16 cores, benchmarked on the infinite XY chain. The calculations are carried out on Nvidia Tesla V100 GPU and Intel Xeon 6230 CPU, with retained bond dimensions up to $D = 360$. The dashed line depicts the scaling $t_h \sim D^4$.

attains very high accuracy, with relative error $\sim 10^{-7}$ at low temperature (down to $\beta \simeq 400$).

GPU acceleration. We implement ∂ TRG with the PYTORCH library, and we take advantage of GPU computing to significantly accelerate the simulations. In Fig. 3(b), we show the elapsed hours t_h versus D in the simulations of an infinite XY chain on GPU and CPU, respectively. To quantify the speedup, t_h is monitored at $\beta = 12.8$, where the computation time falls well into a logarithmic scaling regime versus β , i.e., $t_h \propto \ln \beta$ [38].

From Fig. 3(b), we observe approximately 40 times GPU acceleration (for $D = 360$ calculations), as compared to single-core CPU calculations, and over seven times speedup to the 16-core parallel job. Moreover, in Fig. 3(b), the t_h curves show algebraic scaling versus D , i.e., $t_h \sim D^\gamma$, for sufficiently large D where γ values are found slightly less than 4. These appealing benchmarks, together with previous tests in Ref. [46], suggest that GPU acceleration indeed constitutes a very promising technique to be fully explored in quantum many-body computations, particularly in tensor network simulations.

Thermodynamics of finite-size quantum lattice models. Now we apply ∂ TRG to finite-size chains and cylindrical geometries of finite width W (and length L), and we try to approach the thermodynamic limit by increasing the system size, which has been proved to be very successful in ground-state simulations of quantum frustrated magnets [47,48]. Note that the sweep optimization needs to be adapted when applied to the finite-size systems, i.e., we not only scan $\{w^{(i)}\}$ between different scales but also among different lattice sites/bonds.

For a finite-size system on the 1D or 2D lattice, the high- T density matrix $\rho(\tau)$ can be initialized through a discretization-error-free series expansion technique [38,49]. It has been shown to be preferable, over Trotter-Suzuki-type initializations, in dealing with 2D systems defined on, e.g., long cylinders [16,17]. The benchmark results on the finite-size XY chain can be found in the Supplemental Material

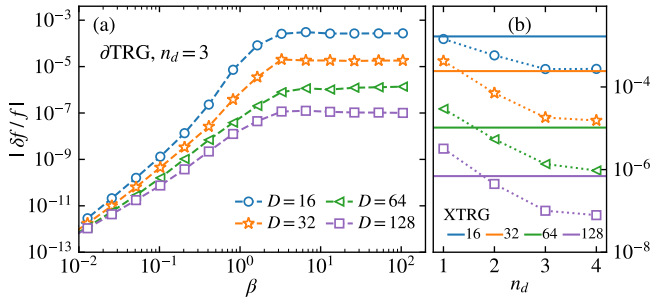


FIG. 4. (a) Relative errors of the transverse-field Ising model on a 4×4 open square lattice, at critical transverse field $h = h_c$. (b) ∂ TRG results with various D are plotted vs optimization depths $n_d = 1, 2, 3,$ and 4 . The comparison is at a fixed low temperature $\beta \simeq 105$, and the standard XTRG results are shown with solid lines.

[38], where deep optimization into n_d layers gains remarkable improvement in accuracy.

As a demonstration for 2D simulations, below we focus on the transverse-field Ising model on a square lattice, i.e.,

$$H = \sum_{(i,j)} JS_i^z S_j^z - h_x \sum_i S_i^x, \quad (3)$$

where $J = -1$ (ferromagnetic) is set as the energy scale. The model undergoes a magnetic order-disorder quantum phase transition at a critical field $h_c \simeq 1.52219(1)$ [50]. Through a snake-path mapping into a quasi-1D lattice [17], the interaction information of the Hamiltonian Eq. (3) on a width W cylinder can be encoded in a compact MPO of bond dimension $D_H = W + 2$ [16,51]. Given the MPO representation of H , ∂ TRG works automatically and produces accurate results as benchmarked below.

First, we run ∂ TRG simulations on a 4×4 square lattice at the critical transverse field $h = h_c$, and we compare the results to exact diagonalization data. In Fig. 4(a), the relative errors $|\delta f/f|$ are plotted versus β . One can observe a high accuracy with an optimization depth $n_d = 3$, which continuously improves upon increasing the bond dimension D . Moreover, to reveal the effects of n_d , in Fig. 4(b) we show $|\delta f/f|$ versus n_d at low temperature, as compared to XTRG data. Indeed the accuracy improves considerably, by orders of magnitude, as n_d increases. For example, the $D = 64, n_d = 4$ ∂ TRG accuracy even matches the $D = 128$ XTRG one.

Large-scale simulations and finite-temperature phase transition. Next, we conduct ∂ TRG calculations of the quantum Ising model on cylinders with various widths W (up to 12) and lengths L . The transverse field is first fixed at $h_x = 1.0 \simeq 2/3h_c$, giving rise to a spin order at low temperature. The long-range order melts at a critical temperature $T_c \simeq 0.42$ through a second-order phase transition [13]. In Fig. 5, we retain only a moderate bond dimension up to $D = 128$ in the calculations, and the optimization depth is up to $n_d = 4$ layers. The internal energy $u(T)$ and magnetic specific heat $c_m(T)$ are computed from the first and second numerical derivatives of $f(T)$, respectively. Following the line developed in XTRG [16,17], we exploit a z -shift technique as well as numerical interpolation to collect dense enough data points and ensure a negligible differential error [38]. Moreover, to eliminate the

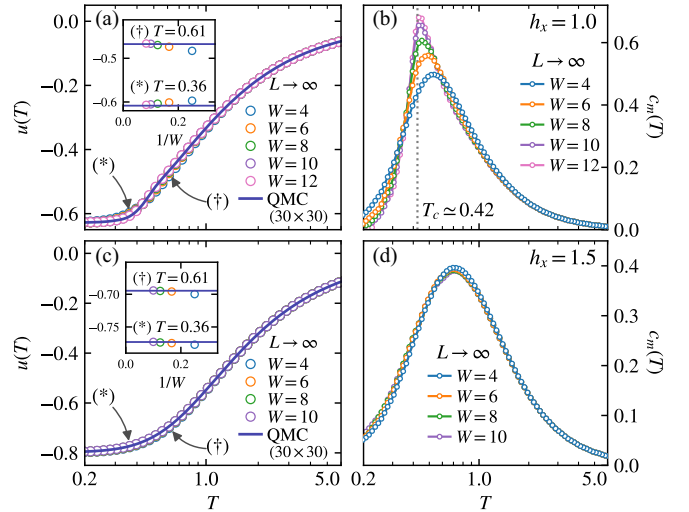


FIG. 5. The plot shows internal energy $u(T)$ and spin-specific heat $c_m(T)$ of the transverse field Ising model, with $h_x = 1$ [(a), (b)] and $h_x = 1.5$ [(c), (d)]. ∂ TRG calculations are run on cylinders of various widths W (with length L extrapolated to infinity). The energy data show excellent agreements with the quantum Monte Carlo (QMC) result [52] on a 30×30 square lattice in both fields. For the $h_x = 1$ case, the peak position in $c_m(T)$ provides an accurate estimate, $\sim 1\%$ relative error, of critical temperature.

finite-length effects, we perform an extrapolation to $L = \infty$ via linear fitting or energy subtraction [38].

Collecting the extrapolated data at each width W , we compare the internal energy $u(T)$ with QMC data in Fig. 5(a). Very good agreement between our cylindrical results and the large-scale QMC data is obtained. The latter are computed on a 30×30 square lattice with fully periodic boundary condition (i.e., torus) that mimics the thermodynamic limit. Furthermore, as shown in the inset, we zoom in at two selected temperatures and find relative errors $|\delta u/u| \sim 10^{-3}$ ($W = 12$ result) with respect to QMC. In Fig. 5(b), by further taking the derivatives of internal energy $u(T)$, we obtain the specific-heat curves $c_m(T)$. It is observed that the peak in $c_m(T)$ gets sharper as W increases, signaling the existence of a phase transition, and the peak locations for wide cylinders are in very good agreement with $T_c \simeq 0.42$ in the thermodynamic limit.

In Figs. 5(c) and 5(d), we provide $u(T)$ and $c_m(T)$ at transverse field $h_x = 1.5 \simeq 0.99h_c$, in close proximity to the quantum phase transition point. Again, $u(T)$ results are in excellent agreement ($|\delta u/u| \sim 10^{-4}$) with the QMC data as shown in Fig. 5(c). It is observed in Fig. 5(d) that the specific heat shows a round peak at around $T/J = 0.7$, which has well converged versus system sizes and does not correspond to any phase transition, which should occur below $T/J = 0.2$.

Conclusion and outlook. Inspired by the essential correspondence between the backpropagation and SRG of tensor networks, we propose the framework of ∂ TRG. With ∂ TRG, we make much better use of tensor parameters by increasing the optimization depth instead of merely enlarging parameter space dimension D . As a result, a moderate D can lead to an unprecedented high accuracy in simulating the thermodynamics of 2D quantum models. ∂ TRG has the same advantage as SRG in that it can optimize both the wave-function

representation and the renormalization transformations, globally and automatically. Therefore, ∂ TRG constitutes a promising tool to investigate very challenging many-body problems, e.g., frustrated antiferromagnets, fermionic Hubbard models, etc., which are currently of great research interest.

Acknowledgments. We are indebted to Jian Cui, Jan von Delft, Yannick Meurice, and Andreas Weichselbaum for helpful discussions. This work was supported by the National

Natural Science Foundation of China (Grants No. 11774420, No. 11834014, No. 11974036, and No. 11774398), the National R&D Program of China (Grants No. 2016YFA0300503 and No. 2017YFA0302900), German Research Foundation (DFG WE4819/3-1) under Germany's Excellence Strategy - EXC-2111 - 390814868 and by the Research Funds of Renmin University of China (Grants No. 20XNLG19). Our code implementation in PyTorch is publicly available [33].

-
- [1] M. Levin and C. P. Nave, Tensor Renormalization Group Approach to Two-Dimensional Classical Lattice Models, *Phys. Rev. Lett.* **99**, 120601 (2007).
- [2] Z.-C. Gu and X.-G. Wen, Tensor-entanglement-filtering renormalization approach and symmetry-protected topological order, *Phys. Rev. B* **80**, 155131 (2009).
- [3] Z. Y. Xie, J. Chen, M. P. Qin, J. W. Zhu, L. P. Yang, and T. Xiang, Coarse-graining renormalization by higher-order singular value decomposition, *Phys. Rev. B* **86**, 045139 (2012).
- [4] G. Evenbly and G. Vidal, Tensor Network Renormalization, *Phys. Rev. Lett.* **115**, 180405 (2015).
- [5] G. Evenbly and G. Vidal, Tensor Network Renormalization Yields the Multiscale Entanglement Renormalization Ansatz, *Phys. Rev. Lett.* **115**, 200401 (2015).
- [6] S. Yang, Z.-C. Gu, and X.-G. Wen, Loop Optimization for Tensor Network Renormalization, *Phys. Rev. Lett.* **118**, 110504 (2017).
- [7] M. Bal, M. Mariën, J. Haegeman, and F. Verstraete, Renormalization Group Flows of Hamiltonians using Tensor Networks, *Phys. Rev. Lett.* **118**, 250602 (2017).
- [8] H. C. Jiang, Z. Y. Weng, and T. Xiang, Accurate Determination of Tensor Network State of Quantum Lattice Models in Two Dimensions, *Phys. Rev. Lett.* **101**, 090603 (2008).
- [9] W. Li, S.-J. Ran, S.-S. Gong, Y. Zhao, B. Xi, F. Ye, and G. Su, Linearized Tensor Renormalization Group Algorithm for the Calculation of Thermodynamic Properties of Quantum Lattice Models, *Phys. Rev. Lett.* **106**, 127202 (2011).
- [10] P. Czarnik, L. Cincio, and J. Dziarmaga, Projected entangled pair states at finite temperature: Imaginary time evolution with ancillas, *Phys. Rev. B* **86**, 245101 (2012).
- [11] Y.-L. Dong, L. Chen, Y.-J. Liu, and W. Li, Bilayer linearized tensor renormalization group approach for thermal tensor networks, *Phys. Rev. B* **95**, 144428 (2017).
- [12] A. Kshetrimayum, M. Rizzi, J. Eisert, and R. Orús, Tensor Network Annealing Algorithm for Two-Dimensional Thermal States, *Phys. Rev. Lett.* **122**, 070502 (2019).
- [13] P. Czarnik and J. Dziarmaga, Variational approach to projected entangled pair states at finite temperature, *Phys. Rev. B* **92**, 035152 (2015).
- [14] P. Corboz, P. Czarnik, G. Kapteijns, and L. Tagliacozzo, Finite Correlation Length Scaling with Infinite Projected Entangled-Pair States, *Phys. Rev. X* **8**, 031031 (2018).
- [15] P. Czarnik and P. Corboz, Finite correlation length scaling with infinite projected entangled pair states at finite temperature, *Phys. Rev. B* **99**, 245107 (2019).
- [16] B.-B. Chen, L. Chen, Z. Chen, W. Li, and A. Weichselbaum, Exponential Thermal Tensor Network Approach for Quantum Lattice Models, *Phys. Rev. X* **8**, 031082 (2018).
- [17] H. Li, B.-B. Chen, Z. Chen, J. von Delft, A. Weichselbaum, and W. Li, Thermal tensor renormalization group simulations of square-lattice quantum spin models, *Phys. Rev. B* **100**, 045110 (2019).
- [18] S. R. White, Density Matrix Formulation for Quantum Renormalization Groups, *Phys. Rev. Lett.* **69**, 2863 (1992).
- [19] Z. Y. Xie, H. C. Jiang, Q. N. Chen, Z. Y. Weng, and T. Xiang, Second Renormalization of Tensor-Network States, *Phys. Rev. Lett.* **103**, 160601 (2009).
- [20] H. H. Zhao, Z. Y. Xie, Q. N. Chen, Z. C. Wei, J. W. Cai, and T. Xiang, Renormalization of tensor-network states, *Phys. Rev. B* **81**, 174411 (2010).
- [21] H.-H. Zhao, Z.-Y. Xie, T. Xiang, and M. Imada, Tensor network algorithm by coarse-graining tensor renormalization on finite periodic lattices, *Phys. Rev. B* **93**, 125115 (2016).
- [22] G. Carleo and M. Troyer, Solving the quantum many-body problem with artificial neural networks, *Science* **355**, 602 (2017).
- [23] J. Carrasquilla and R. G. Melko, Machine learning phases of matter, *Nat. Phys.* **13**, 431 (2017).
- [24] E. P. L. van Nieuwenburg, Y.-H. Liu, and S. D. Huber, Learning phase transitions by confusion, *Nat. Phys.* **13**, 435 (2017).
- [25] E. Stoudenmire, and D. J. Schwab, Supervised learning with tensor networks, in *Advances in Neural Information Processing Systems 29*, edited by D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett (Curran Associates, Inc. 2016), pp. 4799–4807.
- [26] S. Foreman, J. Giedt, Y. Meurice, and J. Unmuth-Yockey, Examples of renormalization group transformations for image sets, *Phys. Rev. E* **98**, 052129 (2018).
- [27] Z.-Y. Han, J. Wang, H. Fan, L. Wang, and P. Zhang, Unsupervised Generative Modeling using Matrix Product States, *Phys. Rev. X* **8**, 031012 (2018).
- [28] C. Guo, Z. M. Jie, W. Lu, and D. Poletti, Matrix product operators for sequence-to-sequence learning, *Phys. Rev. E* **98**, 042114 (2018).
- [29] S.-H. Li and L. Wang, Neural Network Renormalization Group, *Phys. Rev. Lett.* **121**, 260601 (2018).
- [30] M. Koch-Janusz and Z. Ringel, Mutual information, neural networks and the renormalization group, *Nat. Phys.* **14**, 578 (2018).
- [31] H.-J. Liao, J.-G. Liu, L. Wang, and T. Xiang, Differentiable Programming Tensor Networks, *Phys. Rev. X* **9**, 031041 (2019).
- [32] A. Paszke, G. Chanan, Z. Lin, S. Gross, E. Yang, L. Antiga, and Z. Devito, Automatic differentiation in PyTorch, in *Conference on Neural Information Processing Systems (NIPS, Long Beach, CA, 2017)*.
- [33] The official website of PyTorch is at <https://pytorch.org>.

- [34] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, Learning representations by back-propagating errors, *Nature (London)* **323**, 533 (1986).
- [35] D. B. Parker, Learning logic, Report No. TR-47, MIT, Cambridge, MA, 1985.
- [36] Y. LeCun, L. D. Jackel, B. Boser, J. S. Denker, H. P. Graf, I. Guyon, D. Henderson, R. E. Howard, and W. Hubbard, Handwritten digit recognition: applications of neural network chips and automatic learning, *IEEE Commun. Mag.* **27**, 41 (1989).
- [37] Y. LeCun, Y. Bengio, and G. Hinton, Deep learning, *Nature (London)* **521**, 436 (2015).
- [38] See Supplemental Material at <http://link.aps.org/supplemental/10.1103/PhysRevB.101.220409>, where we briefly recapitulate the basic idea of TRG in Sec. A, SRG in Sec. B, backpropagation in Sec. C, and the Quasi-Newton optimization of isometries in Sec. D. Detailed discussions on the correspondence between SRG and backpropagation are provided in Sec. E, and the initialization of $\rho(\tau)$ is in Sec. F. In addition, the analytical solutions (Sec. G), computational hours in an infinite XY chain (Sec. H), finite XY chain results (Sec. I), the z -shift technique (Sec. J), and the energy extrapolation in the 2D transverse-field Ising model (Sec. K) are also presented. The supplemental material includes Refs. [53–56].
- [39] J. Nocedal, and S. J. Wright, *Numerical Optimization*, 2nd ed., Springer Series in Operations Research and Financial Engineering (Springer, New York, 2006).
- [40] See, e.g., <https://github.com/HIPS/autograd>.
- [41] G. Vidal, Entanglement Renormalization, *Phys. Rev. Lett.* **99**, 220405 (2007).
- [42] G. Evenbly and G. Vidal, Algorithms for entanglement renormalization, *Phys. Rev. B* **79**, 144108 (2009).
- [43] Here for the sake of computational cost, we exploit the MERA update for $w^{(i)}$ at the first scale after the environment is obtained through the backward process, while the remaining $w^{(i)}$ on higher layers are update via the HOTRG technique.
- [44] H.-H. Tu, Universal Entropy of Conformal Critical Theories on a Klein Bottle, *Phys. Rev. Lett.* **119**, 261603 (2017).
- [45] L. Chen, H.-X. Wang, L. Wang, and W. Li, Conformal thermal tensor network and universal entropy on topological manifolds, *Phys. Rev. B* **96**, 174429 (2017).
- [46] A. Milsted, M. Ganahl, S. Leichenauer, J. Hidary, and G. Vidal, TensorNetwork on TensorFlow: A spin chain application using tree tensor networks, [arXiv:1905.01331](https://arxiv.org/abs/1905.01331).
- [47] S. R. White and A. L. Chernyshev, Néel Order in Square and Triangular Lattice Heisenberg Models, *Phys. Rev. Lett.* **99**, 127004 (2007).
- [48] Y.-C. He, M. P. Zaletel, M. Oshikawa, and F. Pollmann, Signatures of Dirac Cones in a DMRG Study of the Kagome Heisenberg Model, *Phys. Rev. X* **7**, 031020 (2017).
- [49] B.-B. Chen, Y.-J. Liu, Z. Chen, and W. Li, Series-expansion thermal tensor network approach for quantum lattice models, *Phys. Rev. B* **95**, 161104(R) (2017).
- [50] H. W. J. Blöte and Y. Deng, Cluster Monte Carlo simulation of the transverse Ising model, *Phys. Rev. E* **66**, 066110 (2002).
- [51] H. Li, Y. D. Liao, B.-B. Chen, X.-T. Zeng, X.-L. Sheng, Y. Qi, Z. Y. Meng, and W. Li, Kosterlitz-Thouless melting of magnetic order in the triangular quantum Ising material TmMgGaO₄, *Nat. Commun.* **11**, 1111 (2020).
- [52] B. Bauer *et al.*, The ALPS project release 2.0: open source software for strongly correlated systems, *J. Stat. Mech.: Theor. Exp.* (2011) P05001.
- [53] E. Efrati, Z. Wang, A. Kolan, and L. P. Kadanoff, Real-space renormalization in statistical mechanics, *Rev. Mod. Phys.* **86**, 647 (2014).
- [54] K. G. Wilson, The renormalization group: Critical phenomena and the Kondo problem, *Rev. Mod. Phys.* **47**, 773 (1975).
- [55] L. P. Kadanoff, Variational Principles and Approximate Renormalization Group Calculations, *Phys. Rev. Lett.* **34**, 1005 (1975).
- [56] L. Chen, D.-W. Qu, H. Li, B.-B. Chen, S.-S. Gong, J. von Delft, A. Weichselbaum, and W. Li, Two-temperature scales in the triangular lattice Heisenberg antiferromagnet, *Phys. Rev. B* **99**, 140404(R) (2019).