Efficient evaluation of high-order moments and cumulants in tensor network states

Colin G. West, Artur Garcia-Saez, and Tzu-Chieh Wei

C. N. Yang Institute for Theoretical Physics and Department of Physics and Astronomy, Stony Brook University, Stony Brook,
New York 11794, USA

We present a numerical scheme for efficiently extracting the higher-order moments and cumulants of various operators on spin systems represented as tensor product states, for both finite and infinite systems, and present several applications for such quantities. For example, the second cumulant of the energy of a state, $\langle \Delta H^2 \rangle$, gives a straightforward method to check the convergence of numerical ground-state approximation algorithms. Additionally, we discuss the use of moments and cumulants in the study of phase transitions. Of particular interest is the application of our method to calculate the so-called Binder cumulant, which we use to detect critical points and study the critical exponent of the correlation length with only small finite numerical calculations. We apply these methods to study the behavior of a family of one-dimensional models (the transverse Ising model, the spin-1 Ising model, and the spin-1 Ising model in a crystal field), as well as the two-dimensional Ising model on a square lattice. Our results show that in one dimension, cumulant-based methods can produce precise estimates of the critical points at a low computational cost. The method shows promise for two-dimensional systems as well.

I. INTRODUCTION

The understanding of quantum many-body systems is one of the foremost goals of modern quantum physics. In addition to various analytical approaches, a new set of numerical tools has emerged for this purpose in recent years, based on tensor network representations of such systems [1–4]. These techniques take advantage of the so-called "area law" for entanglement entropy, obeyed by the low-energy eigenstates of gapped Hamiltonians with local interactions. Tensor networks naturally embody this entanglement structure, dramatically simplifying the degrees of freedom required to describe such states. Initially introduced in the context of gapped one-dimensional systems, where they are referred to as "matrix product states" (MPS) [5–7], tensor network methods rose to even greater prominence when it was realized that the celebrated "density matrix renormalization group" (DMRG) technique [8,9] could be reformulated in terms of an MPS [10–13]. Additional algorithms soon followed DMRG, such as "time evolving block decimation" (TEBD) [14], as well as the infinite-system analogs "iDMRG" and "iTEBD" [15,16]. Since then, tensor network methods have also been readily applied to critical systems by means of the "multiscale entanglement renormalization ansatz" (MERA) [17,18], as well as to higher-dimensional systems, where they are generally termed "tensor network states," or "projected-entangled pair states" (PEPS) [2,19–24]. A principle goal of these algorithms is to obtain precise approximations to ground-state wave functions, which can be used for many purposes. For example, one can compute various quantities and observables in an effort to detect phase transitions, a central problem in many-body physics [7,25–28].

In the Landau symmetry-breaking paradigm for phase transitions, one first looks for an operator M whose expectation value $\langle M \rangle$ can serve as the order parameter, i.e., a quantity whose behavior changes sharply across a critical point. When this order parameter is represented by a local operator, it can be computed efficiently on a tensor network state [2]. But while an expectation value is the most straightforward piece of information associated with an operator and a state, there is considerably more information available which one may want to compute. For instance, one may wish to study the higher moments of the operator $\mu_n = \langle M^n \rangle$. A related set of quantities called "cumulants," typically labeled κ_n, is also frequently of interest. An obvious example is the variance of the operator $\langle \Delta M^2 \rangle$, which is simply the second cumulant $\kappa_2 = \mu_2 - \mu_1^2$. Even more important to the search for phase transitions is the so-called "Binder cumulant," first introduced by Binder in 1981 in a study of the classical Ising model [29]. In many settings, such as thermal or disordered systems, it is considered to be one of the most accurate and reliable means of detecting a critical point [30–32], and it has since been applied to a wide variety of models [33–41].

Computing these higher-order moments and cumulants, however, is less straightforward. Direct calculation quickly becomes impractical for large n, since the number of terms to evaluate can be exponential in n. In a classical system with a Hamiltonian H_0, one might define $H(\lambda) = H_0 + \lambda M$, and relate the higher moments of M to the derivatives of an associated partition function, using $\langle M^n \rangle = (\beta \frac{\partial}{\partial \lambda})^n \text{Tr}(e^{-\beta H(\lambda)})$. In quantum systems, however, this equation only holds when $[H_0, M] = 0$, which is not true for a wide variety of physically interesting cases. Because of these barriers to direct calculation, usage of techniques such as the powerful Binder cumulant has in the past been generally confined to studies based on quantum Monte Carlo [33].

The question naturally arises whether these quantities can be efficiently and systematically evaluated using the elegant structure of a tensor network state. Here we demonstrate that the answer is yes. The feasibility of using matrix product states for computing the variance of Hamiltonians has been previously pointed out in the context of DMRG [13,42], and is demonstrated by McCulloch in [43] and [44], via the technique of so-called "matrix product operators" (MPO) [2]. In this work we propose an alternative technique, which gives a simple and efficient method to evaluate all general moments and cumulants for tensor network states, based on moment-generating and cumulant-generating functions. We

demonstrate the calculation of moments and cumulants for finite one-dimensional states, and show that the method can also be used for per-site cumulants in the case of an infinite system. We also show how the techniques naturally generalize to finite systems in higher dimensions. These methods have a variety of useful applications which we demonstrate at length, including the use of the Binder and other cumulants to detect critical points to relatively high precision at a low numerical cost (calculation of the Binder cumulants by MPO methods has also been recently considered in [45] and [46]). We also apply the second cumulant of the energy to examine the convergence of numerical methods based on imaginary time evolution.

This paper is organized as follows: In Sec. II we review moments and cumulants, in particular presenting the Binder cumulant and some of its applications. In Sec. III we very briefly review the MPS formalism and discuss how to compute certain simple expectation values. Section IV demonstrates how to use these expectation values to efficiently compute the moments and cumulants of general operators on an MPS. Section V contains examples of the method as applied to three different spin-chain models (the transverse Ising model, the spin-1 Ising model, and the spin-1 Ising model in a crystal field), as well as a demonstration of the method as applied to a two-dimensional system (the transverse Ising model on a square lattice). Our results are summarized in Sec. VI.

II. MOMENTS, CUMULANTS, AND THE BINDER CUMULANT

A state $|\psi\rangle$ and an operator M collectively imply a probability distribution: the probability density function of ψ in M space. The expectation value $\langle M \rangle$ specifies the central value of the distribution, while the complete set of "Moments" defines the entirety of the shape [47]. The nth moment of the distribution is defined to be $\mu_n = \langle M^n \rangle$; the first moment μ_1 is the expectation value $\langle M \rangle$ itself.

The cumulants of the distribution κ_n form an alternative but equivalent way of specifying its shape. These cumulants contain, in total, the same information as the moments; a complete set of either moments or cumulants completely specifies the distribution. Indeed, the nth cumulant can always be expressed as a polynomial combination of the first n moments, and vice versa [48]. For example, as we have noted above, the second cumulant of the distribution is the distribution's variance defined by

$$\kappa_2 = \mu_2 - \mu_1^2. \tag{1}$$

The third cumulant κ_3 gives the distribution's skewness, and is related to the first three moments by

$$\kappa_3 = \mu_3 - 3\mu_2\mu_1 + 2\mu_1^3. \tag{2}$$

Similarly, the fourth cumulant κ_4 is related to the kurtosis, and is given by

$$\kappa_4 = \mu_4 - 4\mu_3\mu_1 - 3\mu_2^2 + 12\mu_2\mu_1^2 - 6\mu_1^4. \tag{3}$$

Although moments and cumulants are properly defined with respect to a distribution and hence depend on both M and $|\psi\rangle$, when $|\psi\rangle$ is general or clear from context we shall refer to μ_n (κ_n) as the "nth moment (cumulant) of M."

The aforementioned Binder cumulant is a particularly useful quantity in the study of critical points and phase transitions. For some system with some known order parameter M, for example a total magnetization $\sum_j \sigma_j$ or a staggered magnetization $\sum_j (-1)^j \sigma_j$, Binder's cumulant represents a modified version of that parameter's fourth cumulant. Though some slight variations exist in the definition, generally it is given by

$$U_4 = 1 - \frac{\langle M^4 \rangle}{3\langle M^2 \rangle^2}. \tag{4}$$

The utility of the Binder cumulant arises from the special features of its length dependence. The behavior of the Binder cumulant at a critical point depends only weakly on the size of the system, and elsewhere its behavior with respect to the system size differs depending upon the phase. For example, below the critical point in a symmetry-breaking magnetic phase the cumulant will increase with the length of the system, but above the critical point, with symmetry unbroken, it decreases instead. The result is that when curves of the Binder cumulant vs temperature are plotted for various lengths, the critical point is indicated by a simultaneous crossing. Typically, because the behavior at the critical point is already approximately universal, only a set of relatively small system sizes need be considered, eliminating the need for complicated extrapolations of very large systems to the thermodynamic limit.

The Binder cumulant also gives access to the critical exponent of the correlation length by means of traditional finite size scaling techniques in which one seeks to "collapse" the data [29,49]. Up to some small finite size corrections (which become increasingly suppressed as the system size increases), the cumulants show a standard functional form [49],

$$U_4(L,B) = \tilde{U}[L^{1/\nu}(B - B_c)], \tag{5}$$

where ν is the usual critical exponent. A plot of U_4 vs $L^{1/\nu}(B - B_c)$ should therefore appear essentially independent of L, since all of the length dependence has been absorbed into the independent variable of the plot. Hence, B_c and ν can be treated as free parameters, and varied until this length independence is optimized; for example, one could seek to minimize the total absolute square distance between the curves for a variety of lengths L.

III. MATRIX PRODUCT STATES AND EXPECTATION VALUES

A. Finite-length chains

To demonstrate how to efficiently compute quantities such as the Binder cumulant for a system represented by an MPS, we must first review the nature of the MPS representation itself. Consider first a simple one-dimensional spin chain of length L with periodic boundary conditions. As an MPS, this state will be expressed as

$$|\psi\rangle = \sum_s \text{Tr}\big(A_1^{(s_1)} A_2^{(s_2)} \cdots A_L^{(s_L)}\big)|s_1 s_2 \cdots s_L\rangle. \tag{6}$$

In other words, a rank-three tensor "A_j" has been associated with each site "j." One index of this tensor, denoted s_j above, remains free and represents the physical degrees of

freedom at site j. The other two indices, typically termed the "virtual indices," are contracted with the virtual indices of the neighboring tensors A_{j-1} and A_{j+1}. The dimension of these virtual indices, often labeled χ, is called the "bond dimension" of the MPS. The choice of χ represents a numerical parameter which can be adjusted to suit the requirements of the context: smaller values are of course less numerically expensive, but larger values can allow the MPS to more accurately represent the features of the state, particularly in systems with long correlation lengths.

In the one-dimensional case, since A is rank-three, for any fixed value of s_j the two remaining virtual indices simply represent a matrix. For this reason, the labels of the virtual indices are often suppressed, as in Eq. (6), with the contraction represented by the expression $A_{j-1}^{(s_{j-1})} A_j^{(s_j)} A_{j+1}^{(s_{j+1})}$ in the same manner that one would write an ordinary product of matrices.

Let us now consider how to calculate the expectation value of an operator with respect to a matrix product state. Consider first the case of a simple operator which is given by a tensor product of on-site operations. For such an operator, of the form

$$Q = \bigotimes_j Q_j, \tag{7}$$

the expectation value $\langle Q \rangle$ is given by

$$\langle \psi | Q | \psi \rangle = \sum_{s_j, s'_j} \left[\mathrm{Tr}\left(\prod_{j=1}^{L} A_j^{(s_j)} \otimes A_j^{*(s'_j)} \right) \prod_{j=1}^{L} \langle s'_j | Q_j | s_j \rangle \right], \tag{8}$$

or equivalently,

$$= \mathrm{Tr}\left[\prod_{j=1}^{L} \sum_{s_j, s'_j} \left(A_j^{(s_j)} \otimes A_j^{*(s'_j)} \right) \langle s'_j | Q_j | s_j \rangle \right]. \tag{9}$$

From this expression, it is clear that, up to normalization, the expectation value is simply a trace over a set of L "transfer matrices," i.e.,

$$\langle Q \rangle = \frac{1}{\langle \psi | \psi \rangle} \mathrm{Tr}\left(\prod_{j=1}^{L} T_j \right), \tag{10}$$

where the T's are defined as

$$T_j \equiv \sum_{s_j, s'_j} \left(A_j^{(s_j)} \otimes A_j^{*(s'_j)} \right) \langle s'_j | Q_j | s_j \rangle. \tag{11}$$

This procedure is also demonstrated in graphical notation in Fig. 1. The norm of the state can be fixed in a similar fashion, by evaluating a transfer matrix for the special case where $Q_j = \mathbb{1}$.

Tensor products of few-body operators can be handled in a similar fashion by grouping the relevant sites. More general operators are simply evaluated by decomposing them into a sum of tensor products. Considerably more detail on the general process of taking expectation values can be found in the now-extensive body of literature on matrix product states [2,4,50]. For our purposes, however, it will be sufficient to be able to evaluate operators of the simple form in Eq. (7).
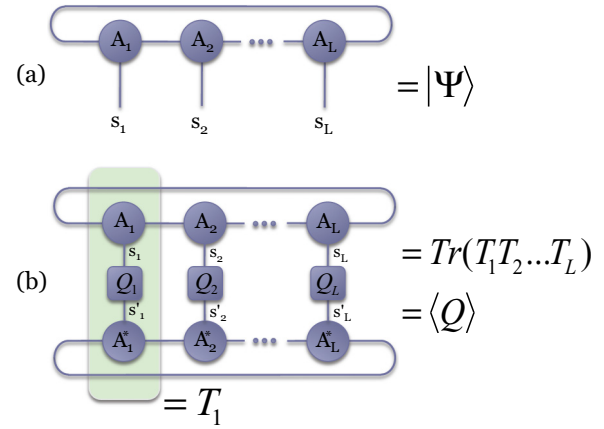

FIG. 1. (Color online) Graphical notation demonstrating the structures of matrix product states. In this notation, a shape represents a tensor, and a line represents an index. Connected lines between shapes represent contracted indices between tensors. (a) A finite spin chain state $|\psi\rangle$ represented as a matrix product state. The state is specified by the set of rank-three tensors $\{A_j\}$, with the physical degrees of freedom s_j left open. (b) The expectation value of a product operator $Q = \otimes_j Q_j$ with respect to $|\psi\rangle$. Each Q_j acts locally on only one site. The total expectation value can be thought of as a trace over a product of transfer matrices T_j, defined in Eq. (11). An example of an individual transfer matrix T_1 is highlighted.

B. Infinite-length chains

One significant advantage of tensor network algorithms is how easily they allow one to directly study certain infinite systems. This can be done for systems with some form of translation invariance, which can be completely represented by their unit cells. For example, consider an one-dimensional infinite system possessing translation invariance with respect to a unit cell of length ℓ. Represented as a matrix product, the state is of the form given by Eq. (6), but with the further restriction that not all tensors A_i are distinct, and instead repeat every ℓ sites. A state with only a one-site unit cell (full translation invariance) can therefore be specified by only a single tensor A,

$$|\psi\rangle = \sum_s \mathrm{Tr}(A^{(s_1)} A^{(s_2)} \cdots)|s_1 s_2 \cdots\rangle. \tag{12}$$

Similarly, a state with two-site translation invariance ($\ell = 2$) specified by two tensors A_1, A_2 and has the form

$$|\psi\rangle = \sum_s \mathrm{Tr}(A_1^{(s_1)} A_2^{(s_2)} A_1^{(s_3)} A_2^{(s_4)} \cdots)|s_1 s_2 s_3 s_4 \cdots\rangle. \tag{13}$$

Of course, because the sums in Eqs. (12) and (13) run over an infinite number of sites, in general there is no way to specify or compute the coefficients. Certain expectation values, on the other hand, may still be expressed as the limit of an infinite product of transfer matrices. It is quite common, for example, to consider the expectation value of an operator with the same translational invariance as the state in question, by looking at the per-site behavior. For our purposes, we will again be concerned with product operators of the form given in Eq. (7). However, we will now restrict ourselves further by imposing

translation invariance on Q. For an infinite system with a unit cell of length ℓ, we shall consider only Q with $Q_j = Q_{j+\ell}$.

In this situation, one can still sensibly define the expectation value as

$$\langle Q \rangle = \frac{1}{\langle \psi | \psi \rangle} \text{Tr} \left(\prod_{j=1}^{\infty} T_\ell \right). \tag{14}$$

Here the transfer matrix T_ℓ is now "enlarged" to represent an entire unit cell of the chain

$$T_\ell \equiv \prod_{j=1}^{l} T_j. \tag{15}$$

In order to approach the infinite case, we shall first examine the case of a finite but very long chain of length L, so that the product in Eq. (14) is limited to L/ℓ terms, i.e.,

$$\langle Q \rangle_L = \frac{1}{\langle \psi | \psi \rangle} \text{Tr} \left(\prod_{j=1}^{L/\ell} T_\ell \right). \tag{16}$$

For L sufficiently large, the product can then be approximated by considering an eigenvalue decomposition of the transfer matrix $T_\ell = U \Lambda U^{-1}$ and inserting it into Eq. (16) [note that, by construction, the transfer matrix as defined by Eqs. (11) and (15) is Hermitian and hence diagonalizable]. By applying the cyclic property of the trace operation, all U matrices can be made to cancel, leaving us only

$$\langle Q \rangle_L = \frac{1}{\langle \psi | \psi \rangle} \text{Tr}(\Lambda^{L/\ell}) \tag{17}$$

or

$$\langle Q \rangle_L = \frac{1}{\langle \psi | \psi \rangle} \sum_{j=1}^{\chi^2} \lambda_j^{L/\ell}, \tag{18}$$

where λ_j are the diagonal elements of Λ, i.e., the eigenvalues of the matrix. At this point, it can be observed that in the infinite limit, only the largest eigenvalue λ_1 will contribute to the sum. In other words, we have simply

$$\langle Q \rangle_L = \frac{1}{\langle \psi | \psi \rangle} (\lambda_1)^{L/\ell}. \tag{19}$$

To fix the norm, we consider a particular transfer matrix $\tilde{T}_\ell$, defined as usual by Eq. (11) for the special case where $Q = \bigotimes_j \mathbb{1}_j$. Then we calculate $\tilde{\lambda}_1$, the largest eigenvalue of $\tilde{T}_\ell$, which satisfies

$$\langle \psi | \psi \rangle = (\tilde{\lambda}_1)^{L/\ell}. \tag{20}$$

Substituting, we have

$$\langle Q \rangle_L = \left(\frac{\lambda_1}{\tilde{\lambda}_1} \right)^{L/\ell}. \tag{21}$$

We then gain access to the per-site behavior by means of a logarithm, which gives

$$\frac{1}{L} \log\langle Q \rangle_L = \log \left(\frac{\lambda_1}{\tilde{\lambda}_1}^{1/\ell} \right). \tag{22}$$
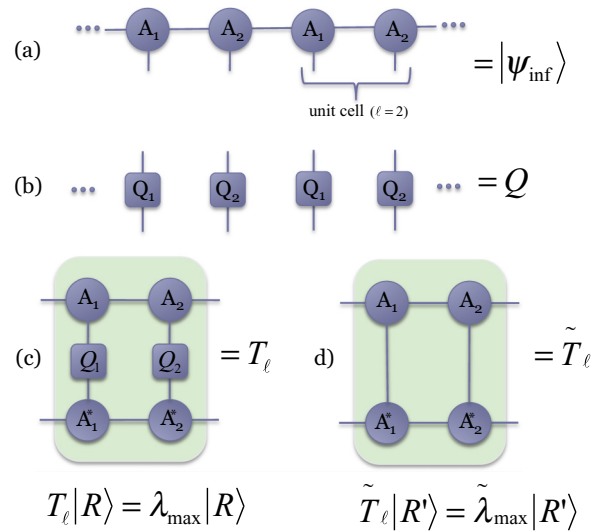

FIG. 2. (Color online) (a) An infinite spin chain state $|\psi_{\text{inf}}\rangle$, possessing translation invariance with respect to a unit cell of length $\ell = 2$, represented as a matrix product state. (b) A product operator $Q = \otimes_j Q_j$ which possesses the same translation symmetry as $|\psi\rangle$, i.e., $Q_j = Q_{j+\ell}$. (c) To compute the quantity of interest, we first construct T_ℓ, the transfer matrix containing an entire unit cell of $|\psi\rangle$ and Q, and extract its dominant eigenvalue λ_1. (d) To normalize the result, we will also need $\tilde{T}_\ell$ (a transfer matrix which contains only the identity operator) and its dominant eigenvalue $\tilde{\lambda}_1$. The desired quantity $\lim_{L\to\infty} \frac{1}{L} \log\langle Q \rangle$ is given by $\log(\lambda_1/\tilde{\lambda}_1)^{1/\ell}$.

Equation (22), however, does not depend on having a finite L. Thus, even for our infinite system, we can consider the limit

$$\lim_{L\to\infty} \frac{1}{L} \log\langle Q \rangle = \log \left(\frac{\lambda_1}{\tilde{\lambda}_1}^{1/\ell} \right). \tag{23}$$

Hence, with these procedures (illustrated graphically in Fig. 2), we can extract information about product operators in their infinite limit even though their expectation values generally diverge. As we will show below, this information will be sufficient to compute the cumulants of operators with translation invariance even in the infinite case.

IV. EVALUATING HIGHER-ORDER MOMENTS AND CUMULANTS

A. Moment and cumulant-generating functions

For a given operator M and a state $|\psi\rangle$, there is an associated function F which contains all of the nonlocal information about the higher moments $\langle M^n \rangle$, and yet, as we shall subsequently demonstrate, can still be efficiently evaluated within the framework of a tensor product state. In particular, this function is given by $F(a) \equiv \langle e^{aM} \rangle$.

In probability theory, $F(a)$ is termed the "moment generating function" of the probability distribution. It is so named because the information about every moment of the distribution is not only contained, but readily accessible from this single function. This can be made explicit by considering a Taylor expansion of e^{aM} about $a = 0$ and then computing

the expectation value in $F(a)$ term-by-term

$$F(a) = 1 + a\langle M \rangle + \frac{a^2}{2}\langle M^2 \rangle + \cdots . \tag{24}$$

From the result, it is clear that every (nonvanishing) moment $\langle M^n \rangle$ will appear in the expansion. Furthermore, these moments can be directly accessed by computing

$$F^{(n)}(a) \equiv \mu_n + O(a), \tag{25}$$

where $F^{(n)}(a)$ is as usual the nth derivative of F.

The moment-generating function F is closely related to the so-called "characteristic function" of the distribution $G(a) \equiv \langle e^{iaM} \rangle$ [51]. For typical states with well-behaved wave functions, these functions will be essentially interchangeable (up to a factor of i). Hence in this work, both will be used, sometimes in combination, depending on the particular moment or operator being computed. It should be noted, however, that for some "pathological" wave functions, such as those specifying a Lorentzian probability distribution, the function $F(a)$ may fail to exist. The characteristic function $G(a)$, however, being the expectation value of a bounded operator, does not suffer from this complication in any situation [52].

While Eq. (25) gives a result for the desired moment which is only accurate up to first order in the parameter a, the precision can be improved by instead computing appropriate combinations of the functions $F(a)$, $F(-a)$, $G(a)$, and $G(-a)$. For example, when seeking to compute an even-ordered moment, i.e., a moment of the form $\langle M^{2n} \rangle$, we construct

$$\tilde{F}(a) = F(a) + F(-a) = 2 + a^2\langle M^2 \rangle + \frac{a^4}{12}\langle M^4 \rangle + \cdots . \tag{26}$$

And hence obtain the desired moments from the relation

$$\tilde{F}^{(n)}(a) \propto \langle M^{2n} \rangle + O(a^2). \tag{27}$$

Odd-ordered moments can of course be found to higher precision from $F(a) - F(-a)$. Even greater precision can also be obtained by including the characteristic functions. The combination $F(a) + F(-a) - G(a) - G(-a)$, for instance, determines $\langle M^2 \rangle$ up to $O(a^4)$.

We employ a similar technique to extract the cumulants of the distribution. This is done by means of the "cumulant generating function," defined as

$$l_F(a) \equiv \log F(a). \tag{28}$$

To see the utility of this function, observe that

$$l_F(a) \approx \log \left(1 + a\langle M \rangle + \frac{a^2}{2}\langle M^2 \rangle + \cdots \right). \tag{29}$$

For small enough values of a, one can see from the expansion $\log(1 + x) \approx x - \frac{1}{2}x^2 + \cdots$ that

$$l_F(a) = a\langle M \rangle + \frac{a^2}{2}\langle M^2 \rangle - \frac{a^2}{2}\langle M \rangle^2 + O(a^3). \tag{30}$$

Grouping these terms by the powers of a, we find that in fact

$$l_F(a) = a\kappa_1 + \frac{a^2}{2}\kappa_2 + \cdots . \tag{31}$$

In other words, the derivatives of the function $l_F(a)$ give us direct access to the cumulants in the same manner as the moments in Eq. (25),

$$l_F^{(n)}(a) = \kappa_n + O(a). \tag{32}$$

Of course, as with the moments, appropriate combinations of $l_F(a)$, $l_F(-a)$, and the associated complex functions can be used to suppress the higher-order terms and improve the accuracy.

B. Evaluating generating functions on a finite matrix product state

Evaluating all of these moments and cumulants thus boils down to evaluating the expectation values of operators like e^{aM}. It remains to be shown that these operators, which we term "moment generating operators," can be applied in an efficient manner. Fortunately, the exponential structure of the operator guarantees that this is indeed the case.

We will consider these operators in two cases, depending on the nature of the operator M. The first, special case is the large class of operators where M can be written as $\sum_j O_j$, where j runs over all the sites in the system and for some arbitrary set of on-site operators $\{O_j\}$. Most usefully, this set of operators contains the traditional magnetization operators such as $M_x = \sum_j \sigma_j^x$, as well as staggered magnetizations, crystal field magnetizations, etc. Subsequently, we will examine the more general case where the terms within M act on more than one site at a time.

1. Sums of single-body operators

In this case, since the operators O_j all act at separate sites, the combined operator M is in fact simply a Kronecker sum $M = \bigoplus_j O_j$. From this it follows that we can write [53]

$$e^{aM} = \bigotimes_j e^{aO_j}. \tag{33}$$

In other words, the moment-generating operator can be decomposed into a set of operators $\{e^{aO_j}\}$, each acting only at a single site. The moment generating function, in turn, is just the expectation value of all these operators applied simultaneously.

Since our moment-generating operator can be written in this tensor product format, we can compute its expectation value directly through Eqs. (10) and (11), with $Q_j = e^{aO_j}$ (see Fig. 3). The calculation of our moment-generating function $F(a)$ has therefore been reduced to the calculation of L transfer matrices and a single trace over their product. In practice, to calculate a higher moment like $\langle M^2 \rangle$, we then need to repeat this procedure and compute F for slightly different values of a, so that it is possible to evaluate the necessary derivative numerically. This can be done through any of the wide variety of standard methods; in this work we have used primarily the classic divided difference formulas [54]. In general, the more values of a at which $F(a)$ is computed, the higher the accuracy of the derivative. However, since the initial a is already chosen to be quite small, in practice it is often the case that only a very small number of points needs to be computed (to check the behavior and accuracy, the procedure can always be repeated with a smaller value of a).
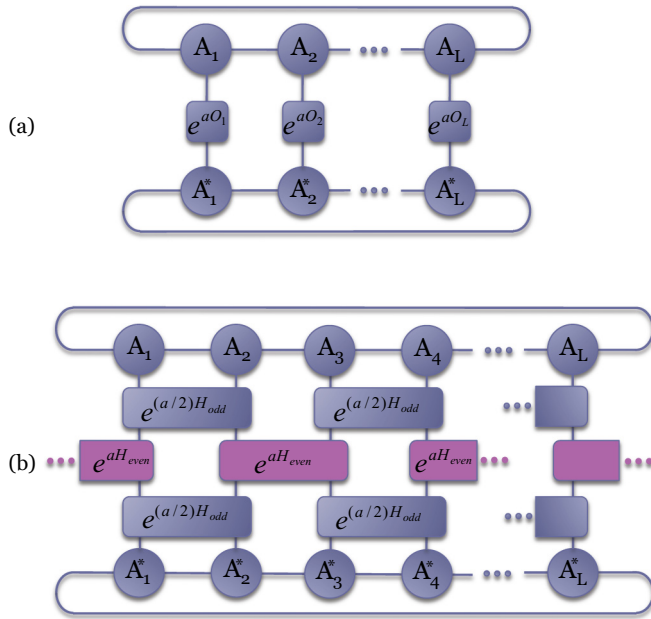
FIG. 3. (Color online) (a) Graphical representation of the moment-generating function $F(a) = \langle e^{aO} \rangle$ for an operator $M = \sum_j O_j$. Since each term in M acts at only one site, the moment-generating operator possesses the same structure, even though the moments M^n are fundamentally nonlocal. (b) The moment-generating function for an operator which is the sum of two-body terms and which possesses the form $H = H_{\text{odd}} + H_{\text{even}}$, such as the transverse Ising Hamiltonian defined in Eq. (38). The operator is approximated by the second-order Suzuki-Trotter formula in Eq. (43), which produces three "layers" of operations. Each layer is a sum of two-body terms.

Let us examine now the performance of the method for the case of $\langle M^2 \rangle$. A second derivative is necessary, which can be computed to second order in a from three values of a, centered at $a = 0$ [54]. Noting that when $a = 0$ we have trivially $F(a) = 1$, it follows that we only have to compute two expectation values, each involving the construction of (at most) L transfer matrices, which are then multiplied together and traced over. By comparison, to compute $\langle M^2 \rangle$ directly, as the sum of all correlators $\langle O_j O_k \rangle$, requires the construction of the same number of transfer matrices (to cover the special case of the correlator where $j = k$), but these matrices must be multiplied and traced over up to L^2 separate times. Since some of these products of transfer matrices in these calculations will appear more than once, the actual computational cost can be reduced somewhat through use of a suitably "dynamically programed" algorithm, where previously calculated products are saved and recycled [2]. Even in this case, however, far more than two solitary products would be required. Furthermore, as the order of the desired moment μ_n increases, the advantage of the moment-generating function method becomes increasingly pronounced, as the numerical derivative will require only approximately n expectation values, instead of L^n.

Simply put, the fact that the exponential nature of the moment generating operator turns long Kronecker sums into simple Kronecker products makes it ideally suited for use with a matrix product state. In all cases, only a small number

of expectation values must be computed in order to allow the calculation of a numerical derivative, with each expectation value containing the operator e^{aO_j} at every site j. Moreover, application of these local operators does not increase the bond dimension of the state.

Such moments can in principle also be computed by means of an MPO [43,44]. As a straightforward demonstration of this, consider for example the MPO given by [55]

$$\hat{C}_j = \begin{pmatrix} \mathbb{1}_j & 0 \\ O_j & \mathbb{1}_j \end{pmatrix}, \tag{34}$$

coupled with the boundary conditions

$$\langle \phi_L | = \begin{pmatrix} 0 & 1 \end{pmatrix} \tag{35}$$

and

$$|\phi_R\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}. \tag{36}$$

To evaluate the moments, one defines the total MPO W to be

$$W \equiv \langle \phi_L | \prod_{j=1}^{L} \hat{C}_j | \phi_R \rangle$$

$$= \sum_{j=1}^{L} O_j.$$

so that

$$\mu_n = \langle \psi | W^n | \psi \rangle. \tag{37}$$

In this naive implementation, each application of W to $|\psi\rangle$ increases the bond dimension of the state by a factor of 2, and thus, to calculate the nth moment in this way requires a bond dimension exponential in n. This increase can be overcome, if necessary, by means of a standard truncation approximation, as done in the TEBD algorithm. Alternatively, a more sophisticated MPO can be constructed [56] which represents the operator M^n, but with a bond dimension of just $n + 1$, resulting in a procedure which still scales linearly.

2. Sums of many-body operators

We now examine the case of calculating the higher-order moments of a more general set of operators $M = \sum_j O_j O_{j+1} \cdots O_{j+k}$. In other words, we consider operators which are a sum of terms acting on at most k sites at a time. So long as k is finite, it remains possible to evaluate the moment generating functions with a single expectation value. This can be done by appealing to the same TEBD technique [14,57] widely used to simulate time evolution and imaginary time evolution in tensor network states.

To begin, we partition the terms of the operator into classes of mutually commuting operators. This will require at most k classes. For example, consider as an operator the two-body transverse Ising Hamiltonian

$$H = -\sum_j \sigma_j^x \sigma_{j+1}^x + B\sigma_j^z. \tag{38}$$

For simplicity, let us write H in a manner which makes it explicitly a sum of two-body operators

$$H = -\sum_j \sigma_j^x \sigma_{j+1}^x + \frac{B}{2}\left(\sigma_j^z + \sigma_{j+1}^z\right). \tag{39}$$

Then, the terms can be partitioned between the even and odd pairs of sites, and H can be written as

$$H = H_{\text{even}} + H_{\text{odd}}, \tag{40}$$

with

$$H_{\text{even}} = -\sum_{j \text{ even}} \sigma_j^x \sigma_{j+1}^x + \frac{B}{2}\left(\sigma_j^z + \sigma_{j+1}^z\right) \tag{41}$$

and

$$H_{\text{odd}} = -\sum_{j \text{ odd}} \sigma_j^x \sigma_{j+1}^x + \frac{B}{2}\left(\sigma_j^z + \sigma_{j+1}^z\right). \tag{42}$$

The moment generating function therefore has the form $F(a) = e^{aH_{\text{even}} + aH_{\text{odd}}}$, and admits a Suzuki-Trotter approximation [58]. To second order in a, this has the form

$$e^{aH_{\text{even}} + aH_{\text{odd}}} \approx e^{\frac{a}{2}H_{\text{even}}} e^{aH_{\text{odd}}} e^{\frac{a}{2}H_{\text{even}}}. \tag{43}$$

Higher-order versions of the approximation have also been well documented and can be easily substituted where greater precision is required [59].

Because H_{even} and H_{odd} were explicitly constructed to be sums of mutually commuting terms, each exponential on the right-hand side of Eq. (43) is now in the same Kronecker sum form as we had in the case of on-site operators, and each can therefore be equivalently expressed as a single tensor product of operations acting on the entire state at once, in the manner of Eq. (33). This is graphically depicted in Fig. 3. Hence, by applying the three exponentials from Eq. (43) in sequence, we can easily calculate the expectation value that represents the moment-generating function.

From this point, evaluating the expectation value is no different than the case of on-site operators. The bulk of the numerical costs are therefore essentially the same for both the on-site and many-body operator, with only one difference: applying these layers of exponential operators will increase the bond dimension of the system, and the size of these bonds may need to be "truncated" by some approximation scheme to keep the system numerically tractable. This, however, is a common practice in the field of MPS algorithms, and is easily done by means of a Schmidt decomposition (see for example [16]).

While our example considered an operator with $k = 2$, that is, two-body interactions which could be partitioned into two internally commuting classes, the technique easily generalizes to larger interactions. The Suzuki-Trotter approximations, for example, can be iteratively applied to an operator e^{A+B+C} by first approximating $e^{A+(B+C)}$ in terms of e^A and e^{B+C}, and then approximating e^{B+C}.

The ability to calculate moments and cumulants for an operator with many-body terms has a particularly useful application in the world of numerical state estimation. A common goal of tensor network algorithms is the calculation of an approximate numerical ground state, from a given Hamiltonian H. These algorithms are typically iterative in nature, gradually refining the approximation as the energy

E tends towards E_0. It is therefore often important to have a means of actively checking this convergence during the course of the algorithm. The variance (second cumulant) of the Hamiltonian $\langle \Delta H^2 \rangle = \langle H^2 \rangle - \langle H \rangle^2$ is well suited to this task [2]. For $\epsilon = \sqrt{\langle \Delta H^2 \rangle}$, there will be an exact eigenvalue E_{ex} within ϵ of the approximate energy E. In other words

$$|E - E_{\text{ex}}| \leqslant \epsilon. \tag{44}$$

In the case of finite systems, the quantity ϵ can be directly computed from the methods above, and can therefore be directly used as an error bound on the calculated energy of the system. For the infinite case, there is a small subtlety: one does not compute the total energy of the system, since this is infinite, but rather the energy per site E/L. Consequently, a proper "error bound" on the measurement would not be ϵ, but $\epsilon/L = \sqrt{\langle \Delta H^2 \rangle}/L$. This quantity, unfortunately, is inaccessible using the techniques above, since the cumulant generating function on the infinite system gives only $\langle \Delta H^2 \rangle/L$. However, one would still expect the latter number to be monotonically related to the true error. Alternatively, as we will discuss in the next section, the related quantity $\sqrt{\langle \Delta H_I^2 \rangle}/\ell$ can be evaluated on a finite interval of length ℓ. Hence in either case, we can derive something which gives a sufficient criterion for halting a converging algorithm like TEBD or iTEBD. Although ϵ (in the finite case) or $\langle \Delta H^2 \rangle$ (for the infinite) are not guaranteed to be small as soon as the energy has converged, if we iterate our algorithms until they *are* very small, we can be assured that the approximate ground-state energy is very close to the true value E_0.

In [2] a dynamically programed algorithm was given for computing $\langle \Delta H^2 \rangle$ in the context of a finite matrix product state. The method presented here performs at least as efficiently as that case, with the advantage that it can also be applied to infinite (or indeed, higher-dimensional) systems. In [43], the same quantity was presented and evaluated by means of an MPO. As discussed above, the MPO technique can in principle be used as an alternative method to compute other moments and cumulants as well, at the cost of allowing the bond dimension to increase. The MPO techniques will also have an advantage in the case of an operator with nonlocal characteristics or other properties that make Suzuki-Trotter decomposition impractical.

For the case of the energy cumulant, our method is also particularly well suited for use with the TEBD/iTEBD algorithms. We have previously remarked that for a Hamiltonian H, the calculation of the associated moment-generating operator e^{aH} is essentially identical to an imaginary time-evolution operator with a time step of $\delta t = a$. Each iteration of this algorithm therefore amounts to calculating $e^{aH}|\psi\rangle$, from which the moment-generating function $F(a) = \langle e^{aH} \rangle$ can easily be computed. If one also computes $F(-a) = \langle e^{-aH} \rangle$, the error bound can therefore be computed very efficiently up to order a^2 in accordance with Eqs. (26) and (27).

Performing this check at regular intervals throughout the evolution offers a halting condition to certify convergence of the energy. In some respects, this convergence criterion is superior to the typical methods, which often signal a halt when δE, the change in the approximate energy between two successive iteration steps, drops below some minimum

value. Such a method can occasionally give a false sense of convergence when the algorithm "stalls out" and begins evolving only very slowly, despite remaining some distance from the ground state. The variance of the energy provides information not about the convergence of the algorithm but of the energy itself, by identifying when the system is very close to an exact eigenstate (however, in some cases care must be taken that the nearby eigenstate is in fact the ground state, and not some excitation). More details about convergence schemes can be found in Appendix B.

C. Evaluating generating functions on an infinite matrix product state

At first glance it may seem that moment-generating techniques discussed above cannot be applied to infinite systems, since the value of a quantity like $M = \sum_j^\infty O_j$ is clearly diverging, and only related limits like

$$\langle M \rangle = \lim_{L \to \infty} \frac{1}{L} \sum_j^\infty \langle O_j \rangle \qquad (45)$$

are well defined. In this situation, however, while the moment-generating function F defined above may diverge, one can still define and calculate the related quantity

$$F_\infty = \lim_{L \to \infty} \langle e^{aM} \rangle^{1/L}. \qquad (46)$$

As discussed in Sec. III, quantities of this form can in fact be computed quite naturally. Using Eq. (21) (see Fig. 2), clearly we have

$$F_\infty = \lim_{L \to \infty} \langle e^{aM} \rangle^{1/L} = \frac{\lambda_1}{\tilde{\lambda}_1}. \qquad (47)$$

In other words, the desired quantity is simply the largest eigenvalue of the transfer matrix associated with the state and operator in question. Hence by defining

$$l_{F_\infty}(a) = \log F_\infty(a), \qquad (48)$$

we find that we have access to the per-site limits of the cumulants even in the infinite case. In the same manner as with finite systems, they are given by the derivatives of l_{F_∞} with respect to a,

$$l_{F_\infty}^{(n)}(a) = \lim_{L \to \infty} \frac{1}{L} \kappa_n + O(a). \qquad (49)$$

We shall comment briefly on some practical considerations that are important when evaluating l_{F_∞} for a real matrix product state. First, algorithms for generating the states which are based on the iTEBD principle are likely to require the use of a two-site unit cell, even if the state is expected to possess only one-site translation invariance, as a result of the two-body nature of most parent Hamiltonians. In this case, of course, a two-site transfer matrix is required ($\ell = 2$), and we must take a square root of its largest eigenvalue in order to recover the correct per-site limit.

Additionally, we observe that for the second cumulant in particular, it can be particularly desirable to calculate using the characteristic function $G = \langle e^{iaM} \rangle$ instead of F. This is

because one can then take advantage of the fact that

$$\lim_{L \to \infty} \frac{1}{L} \kappa_2 = l_{G_\infty}(a) + l_{G_\infty}(-a) + O(a^2)$$

$$= \log G_\infty(a) + \log G_\infty(-a).$$

Then, combining the two log terms and observing that $G_\infty(-a) = G_\infty(a)^*$, we have

$$\lim_{L \to \infty} \frac{1}{L} \kappa_2 = \log[|G_\infty(a)|^2] + O(a^2). \qquad (50)$$

In other words, we can calculate the per-site second cumulant up to second order in a by evaluating $G_\infty(a)$ only once, and without directly computing any numerical derivative.

Note that, although this method gives access to the per-site limit of cumulants on an infinite system out to arbitrary order, the typical definition of the Binder cumulant, as a rational function of moments, cannot be computed in this way without encountering a divergence. When the Binder cumulant is being studied for its finite-size scaling properties, this difficulty is often immaterial. In another case, however (such as when the limiting value of the Binder cumulant itself is the quantity of interest), we refer the reader to a recent and inventive MPO technique which can be used to access the cumulant directly, see [60,61].

It should be noted also that within a system the moment and cumulant methods can also be applied to any finite subset of spins, for example, $S_I \equiv \sum_{i \in I} S_i^z$, where I is any set of locations. In the infinite case, it can be applied to any subset of spins in a finite interval. For example, one can apply gates $e^{iaS_1^z} \cdots e^{iaS_L^z}$ to only a segment of L sites in the infinite system. The expectation value of this, which we denote by $f_L(a)$, can be straightforwardly calculated by the transfer matrix method, in a manner analogous to the case of a finite state:

$$f_L(a) = \langle v_l | T_L^{S^z} | v_r \rangle / \tilde{\lambda}_1^L, \qquad (51)$$

where as before $\tilde{\lambda}_1$ is the largest eigenvalue of the transfer matrix containing only the identity T_L and $T_L^{S_z}$ is the generalized transfer matrix containing the operator S_z at each site.

Of particular interest in this case, for example, are the moments given by

$$\left\langle \left(\sum_{i=1}^L S_i^z \right)^2 \right\rangle_L = -f^{(2)}(0) \qquad (52)$$

and

$$\left\langle \left(\sum_{i=1}^L S_i^z \right)^4 \right\rangle_L = f^{(4)}(0), \qquad (53)$$

from which the L-dependent Binder cumulant $U_4(L)$ can be computed as

$$U_4(L) = 1 - f^{(4)}(a = 0) / [3 f^{(2)}(0)]^2. \qquad (54)$$

This is the Binder cumulant for a segment of L spins in an infinite chain.

Although our example discussed an operator which was a sum of single-body terms, the technique is not so limited. In a similar fashion, one can compute the moments and cumulants of some finite subset of the Hamiltonian $H_I = \sum_{i \in I} h_i$. In this

case, one would split the chain into two disjoint segments, I and its complement. Then, in the same manner by which a single step of the iTEBD algorithm was used to apply required operator e^{iaH} in the infinite case, so too can the finite TEBD algorithm be used to apply e^{iaH_I} to the finite interval. After this, the interval is recombined with the infinite remainder of the state, and standard finite and infinite MPS techniques can be used to evaluate the requisite expectation value.

D. Higher-dimensional states

Although in this work we shall be applying these techniques to one-dimensional systems, there is nothing about the procedures above that cannot be immediately generalized to finite-sized higher-dimensional states. Consider for example a total magnetization-type operator on a two-dimensional, $L \times L$ square lattice, given by

$$M = \sum_j \sum_k O_{jk}, \tag{55}$$

where O_{jk} represents a specific operator acting locally on site (j,k) of the lattice. Of course by representing both j and k by some composite index J (now running from 1 to L^2) we can immediately see that M is no different than the magnetizationlike operators we considered in the one-dimensional case

$$M = \sum_J O_J, \tag{56}$$

and hence that our earlier analysis goes through: the moment generating operator e^{aM} can still be written as

$$e^{aM} = \bigotimes_J e^{aO_J}. \tag{57}$$

This object is still an operator that acts only locally and whose expectation value can be evaluated all at once (graphically depicted in Fig. 4).

If we consider instead an operator H which contains many-body terms (but for whom each term acts nontrivially only on a finite number of sites), one can play the same tricks as in one dimension, first partitioning the terms into mutually commuting sets of terms $H_1, H_2 + H_3 \cdots$, then expressing the moment-generating operator as

$$e^{aH} = e^{a(H_1 + H_2 + H_3 \cdots)}, \tag{58}$$

and finally applying some form of Suzuki-Trotter approximation as described above to express the operator as a product of exponentials, each of which can be applied to the state all at once. For nearest-neighbor interaction terms on a square lattice, the procedure is essentially identical to the one-dimensional case, except that one must use four classes instead of two: two for interactions in the horizontal direction, and two for the vertical. For operators with more complicated terms (such as a sum of "plaquettes"), the number of partitions may be larger, but in general the costs do not grow rapidly despite the increase in system dimension.

Hence, in either case, the associated moment-generating operators can be disentangled into tensor products or sequences of tensor products, even in higher dimensions. Simply put, the essential "power" of the moment-generating function method
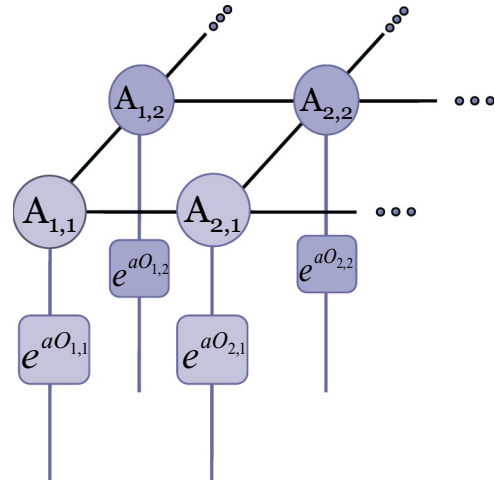

FIG. 4. (Color online) The moment-generating operator e^{aM} for an operator of the form $M = \sum_j \sum_k O_{jk}$, applied to a two-dimensional state on a square lattice. As in the one-dimensional case, the locality of each term in M ensures the locality of the terms in e^{aM}, and hence, the moment-generating operator can still be evaluated all at once, by applying the appropriate on-site operator at each lattice site.

is the fact that the moment-generating operator e^{aM} of a local operator M is itself a local operator, and this fact is unchanged regardless of the dimensionality of the system.

Once the moment-generating operator has been expressed as a tensor product and applied to the state, it still remains to numerically contract the tensor network. It is at this stage where things become more difficult than in the one-dimensional case, since computing the expectation value of *any* operator on a higher-dimensional tensor network state can be quite hard. Exact calculation has been shown to be exponentially costly in L (in particular, it is a #P-hard problem [62]). Nevertheless, a wide variety of numerical techniques have been developed to approximate these contractions efficiently with minimal errors, the details of which are outside the scope of this paper (see for example Refs. [20,63–67]). We do caution, however, that in our experience, the higher the order of the moment, the higher the sensitivity of the result to the errors introduced by approximate contraction.

V. EXAMPLES

A. Spin-1/2 transverse Ising model

As has become almost customary, we begin by demonstrating our technique in the context of the widely studied transverse Ising model; a chain of spin-1/2 particles governed by the Hamiltonian

$$H = -\sum_j^L \sigma_j^x \sigma_{j+1}^x + B\sigma_j^z. \tag{59}$$

This model is a useful proving ground as it has been extensively studied and admits a well-known analytical solution [68,69], as well as possessing a straightforward order parameter of $M_x = \sum_j \sigma_j^x$, the total magnetization in the x direction. We can therefore test our techniques by using
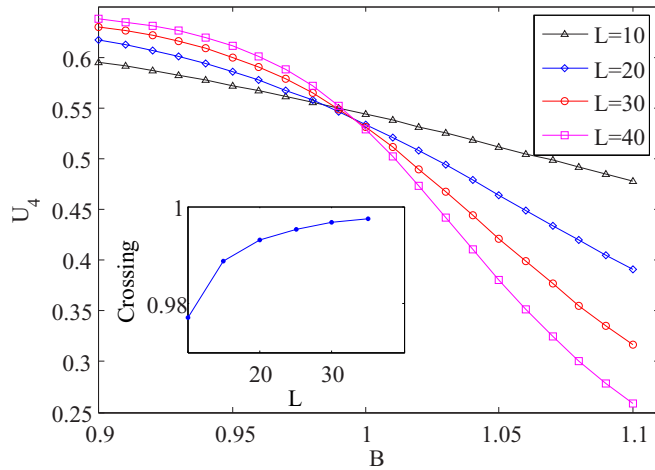
FIG. 5. (Color online) A Binder cumulant study of the transverse Ising model. The cumulants are computed for different system sizes across a range of values for the transverse field B (some intermediate system sizes have been suppressed for clarity of the figure). Crossing points are interpolated for successive pairs of curves, i.e., $L = 10$ and $L = 15$. These crossing values can then be seen to approach the known value of the critical field $B_c = 1$ (inset). The BST algorithm is used to extrapolate these values to the infinite limit, which gives $B_c = 1.001(1)$.

FIG. 6. (Color online) Per-site value of the second cumulant of the longitudinal magnetization $\frac{1}{L}\langle\Delta M_x^2\rangle = \frac{1}{L}(\langle M_x^2\rangle - \langle M_x\rangle^2)$, for the transverse Ising model. The cumulant is plotted for various finite system sizes, plotted against a range of applied fields. As the system length increases, the behavior tends towards the infinite limit (inset). In the limit, the cumulant diverges at the critical point.

them to study the phase transition known to occur exactly at $B_c = 1$. To apply the Binder cumulant technique, we first use a numerical method to find the ground state by solving the generalized eigenvalue problem [2]. We consider system lengths between $L = 10$ and $L = 45$ in steps of five, using a bond dimension of $\chi = 10$ (we verify that increasing the bond dimension does not change the results of the methods up to our working precision). For each system length, ground states of the Hamiltonian in Eq. (59) are computed as we sweep over a range of values for the field coefficient B. Then for each value of L and B, we compute the Binder cumulant using the methods described above, by first computing μ_2 and μ_4 by means of the moment-generating function.

As shown in Fig. 5, the crossings of the Binder cumulants at various lengths are already clustered very close to the transition point, even though the lengths of the states are relatively short compared to the thermodynamic limit. But the location of the critical point can be computed to even greater accuracy by considering the pattern of successive crossings. These crossings show a clear trend towards a limiting value as the system sizes increase. This limiting value can be estimated by means of the BST algorithm [70], which has been found to be a very powerful tool for estimating the infinite limit of a series of data based on finite size corrections which obey a power law, even from a relatively small number of data points [71]. From this extrapolation, we estimate a critical point of $B_c = 1.001(1)$. Here and elsewhere, the reported uncertainty in our extrapolation represents an estimate of the typical uncertainties in the location of the crossing points, propagated through the BST algorithm. Much more detail about the BST technique can be found in Appendix A.

The critical point of the Ising model can be probed directly through the higher-order cumulants of the order parameter, as
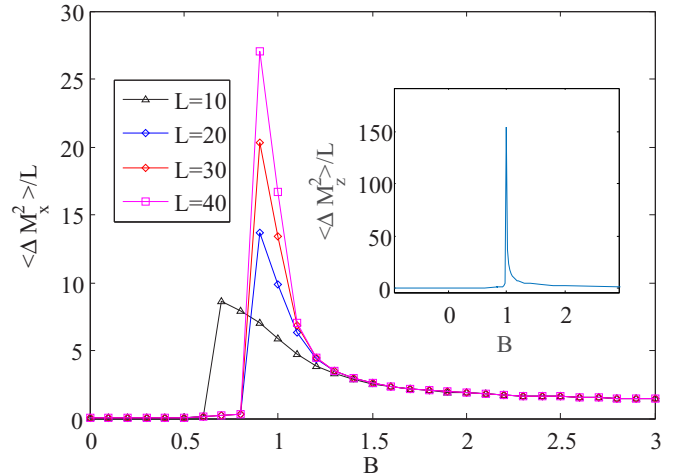
well. Through the techniques above, these can be calculated from the finite systems at various system sizes. Alternatively, we can calculate a ground state for the infinite system through the iTEBD algorithm (in this case using a bond dimension of $\chi = 20$) and then calculating the second cumulant directly. Both procedures are showcased in Fig. 6. The behavior in the infinite case can be seen to agree with the limiting trend of the finite systems as the length is increased. The cumulant can be seen to become singular near the critical point, and can also be used to detect the transition. Using this method we estimate $B_c = 1.00(1)$.

This technique can also be applied to the magnetization in the transverse direction, $M_z = \sum_j \sigma_j^z$. In this case, it is the derivative of the cumulant which becomes singular in the thermodynamic limit to signify the critical point. Again, the results from the finite chains can be seen trending towards the infinite limit. In this case, the expected behavior in the thermodynamic limit can also be calculated analytically in this case by using the known form of the Ising model wave functions. The cumulant as a function of the applied field B is found to be

$$\langle\Delta M_z^2\rangle/L = \begin{cases} 1 & : B \leqslant 1 \\ 1/B^2 & : B > 1 \end{cases}. \qquad (60)$$

This numerical data (see Fig. 7) are in agreement with this expression to a very high degree of accuracy.

The critical exponent of the correlation length of the model can also be studied by means of the Binder cumulant. Once the critical point has been estimated, the curves of U_4 can then be plotted against $L^{1/\nu}(B - B_c)$ for various values of ν. At the true critical exponent, the data should "collapse" to a single functional form independent of L, as seen in Fig. 8.

Additionally, we can use this model to consider the scaling of the Binder cumulant with the correlation length, by employing the procedure outlined above to compute moments and cumulants on a finite subblock of an infinite state. From
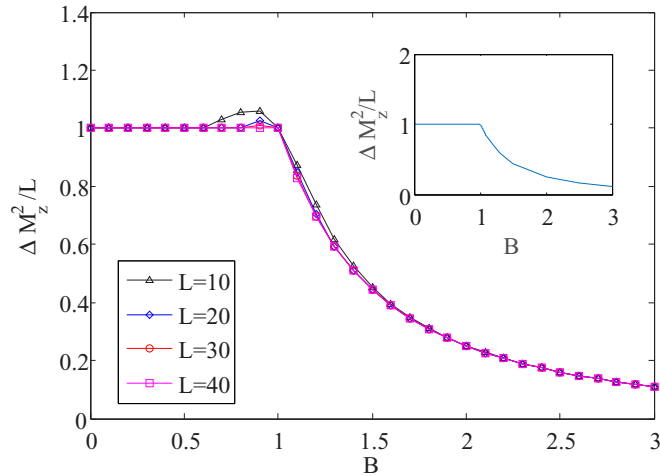
FIG. 7. (Color online) Second cumulant of the transverse magnetization $\langle \Delta M_z^2 \rangle = \langle M_z^2 \rangle - \langle M_z \rangle^2$, for the transverse Ising model (computed per site). The cumulant is plotted for various finite system sizes, plotted against a range of applied fields. As the system length increases, the behavior tends towards the infinite limit (inset) where the derivative of the cumulant shows a discontinuity at the critical point. This behavior is in excellent agreement with the analytical result for the thermodynamic limit, that $\langle \Delta M_z^2 \rangle / L = 1$ for $B < 1$ and $1/B^2$ for $B > 1$.

this, the so-called "critical Binder cumulant" U_4^* (the limiting value of the cumulant at the critical point) can be extracted. As suggested by McCulloch [72], we can evaluate $U_4(L)$ close to the critical point with the choice that $L \approx 10\xi$ for a number of bond dimensions χ. Choosing a subblock of this size should allow us to see the desired scaling behavior while suppressing the finite-χ corrections to the cumulant. This can
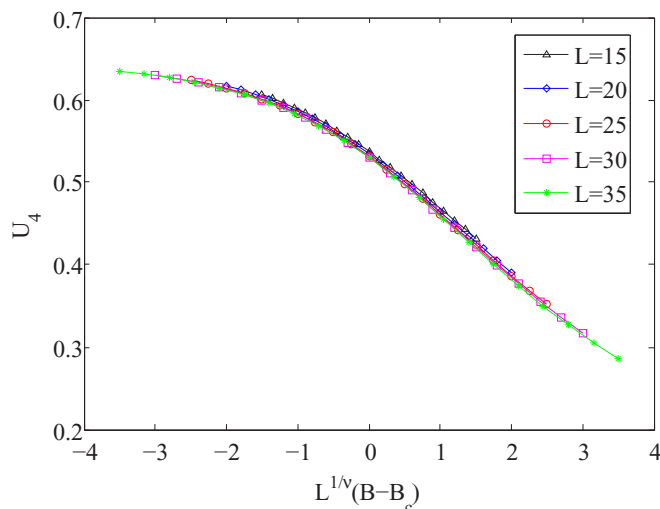
FIG. 8. (Color online) The Binder cumulants for the transverse Ising model, plotted for a variety of system sizes as a function of $L^{1/\nu}(B - B_c)$, for the known values $\nu = 1$ and $B_c = 1$. As expected, for these values the curves are seen to collapse to a functional form essentially independent of the length scale. This property can be used to estimate the values of the critical point and the critical exponent by treating them as fit parameters and optimizing the collapse.
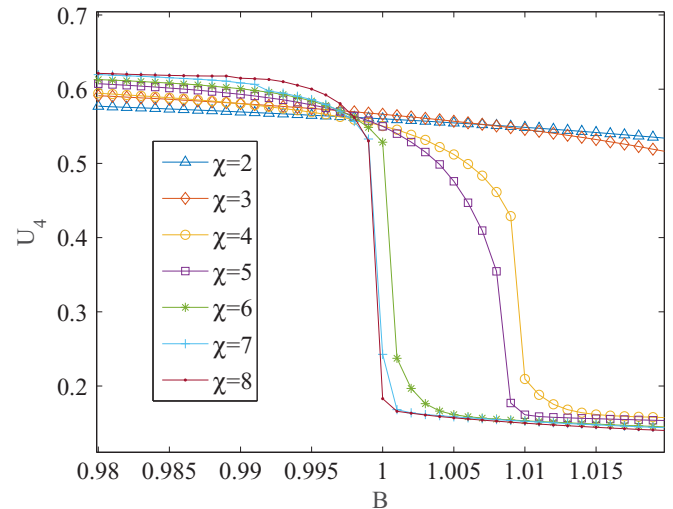

FIG. 9. (Color online) The Binder cumulants for a finite subblock of the infinite transverse Ising model, plotted for a variety of bond dimensions χ. For each bond dimension, the apparent correlation length of the ground-state ξ is used to set the size of the block. The scaling behavior of these blocks as χ increases gives an alternative method for identifying the critical point, as well as allowing an estimate of the critical Binder cumulant ($B_c = 0.998$ and $U_4^* = 0.57$ for this data).

then be computed over a range of transverse field values B, and the resulting Binder cumulant vs B curves for different bond dimensions should cross approximately near the critical point and at approximately the value of U_4^* [72].

A simple demonstration of this, using bond dimensions from $\chi = 2$ to $\chi = 8$, can be seen in Fig. 9. Despite the fact that the bond dimensions are so small, this alternative scaling already gives a crossing of $B_c \approx 0.998$. The value of the cumulant at the crossing point (the so-called "critical Binder cumulant") is $U_4^* \approx 0.57$.

Finally, we can use this model on a finite lattice to demonstrate the utility of the energy variance $\langle \Delta H^2 \rangle$ in assessing numerical convergence, as described above. Starting from a random state with $L = 10$, we apply the TEBD algorithm with the transverse Ising Hamiltonian and evolve towards the ground state, over a range of field strengths B. As shown in Fig. 10, the results are initially somewhat noisy when compared to the analytical E vs B curve, which is reflected by the large error bars computed from $\langle \Delta H^2 \rangle$. However, as the algorithm continues, these error bars shrink and eventually become essentially zero, signaling the complete convergence of the energies.

B. Spin-1 transverse Ising model

We consider next the spin-1 generalization of the Ising model, with Hamiltonian

$$H = -\sum_j^L S_j^x S_{j+1}^x + B S_j^z. \tag{61}$$

Here we have simply replaced the spin-1/2 Pauli matrices from Eq. (59) with their spin-1 counterparts. This model is
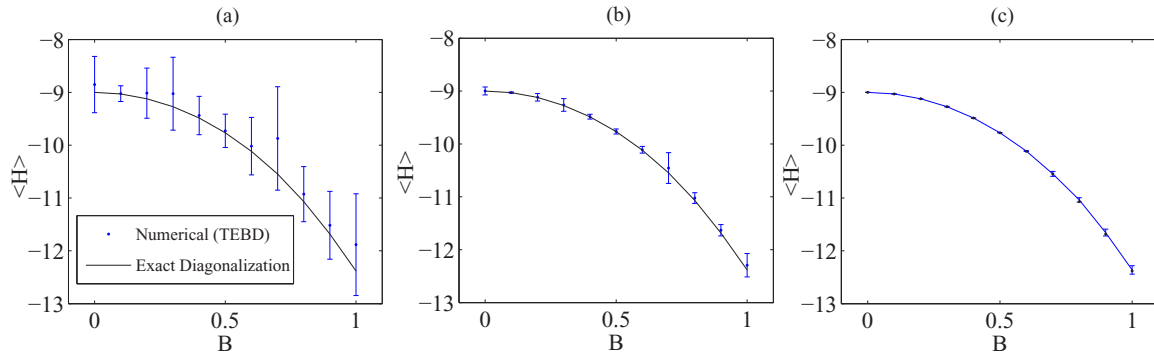
FIG. 10. (Color online) The energy of the spin-1/2 transverse Ising model on a chain of length 10 is calculated using approximate ground states generated by the TEBD algorithm ($\chi = 20$). The numerical data (points) are plotted alongside the exact solution (line). Error bars are calculated from $\epsilon = \sqrt{\langle \Delta H^2 \rangle}$, with $\langle \Delta H^2 \rangle$ the second cumulant of the energies. (a) After 10 steps, the energies are still noisy and the error bars are quite large. (b) After 100 steps, the error bars have clearly decreased, and are largest for the points with the largest discrepancies from the exact solution. (c) By 1000 steps, most error bars are within the size of the data points, and the approximate energies are very close to the known analytical result.

of interest because unlike the spin-1/2 case, it has no exact analytic solution. Nevertheless, in the thermodynamic limit the magnetization is qualitatively similar to the spin-1/2 case. Notably, it still displays a quantum phase transition at a critical value of the transverse field, which has been studied by various numerical methods [73,74]. The accepted value for this critical field is $B_c = 1.326$ [73].

We study this transition point with the same techniques as before, calculating the Binder cumulants from the second and fourth moments of the x magnetization for various system sizes (Fig. 11). The ground states are calculated using the same finite MPS technique and with a bond dimension of $\chi = 20$. The successive crossings are then compared and a limiting value extrapolated using BST. In this case, we compute an estimate of the transition at $B_c = 1.327(1)$.
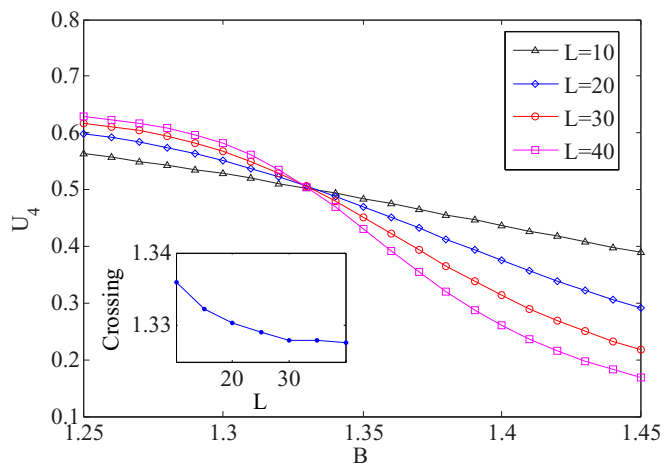

FIG. 11. (Color online) A Binder cumulant study of the spin-1 transverse Ising model. As above, the cumulants are computed for different system sizes across a range of values for the transverse field B (some intermediate system sizes have been suppressed for clarity of the figure). Crossing points are interpolated for successive pairs of curves, i.e., $L = 10$ and $L = 15$, and the BST algorithm is used to extrapolate these values to the infinite limit, which gives $B_c = 1.327(1)$.

Once again it is also constructive to consider the second cumulant on its own. Numerical calculations of the magnetization for this model invariably show some finite size effects around the transition, producing a finite "tail" near the transition point, which makes an exact determination difficult using the order parameter alone [75]. But the transition appears much more sharply as a singularity when we consider the second cumulant, as in Fig. 12 (higher cumulants such as κ_4 can also be used for this purpose). From this quantity we obtain $B_c = 1.324(2)$, an estimation which is to within less than 0.2%.

As before, we can also study the critical exponent ν, known for this model to be the same as the spin-1/2 case, $\nu = 1$. As a proof of principle, the "data collapse" for the known values of ν and B_c are shown in Fig. 13. As expected, for these values the curves are seen to collapse to a functional form essentially independent of the length scale. This property can be used to estimate the values of the critical point and the critical exponent by treating them as fit parameters and optimizing the collapse.
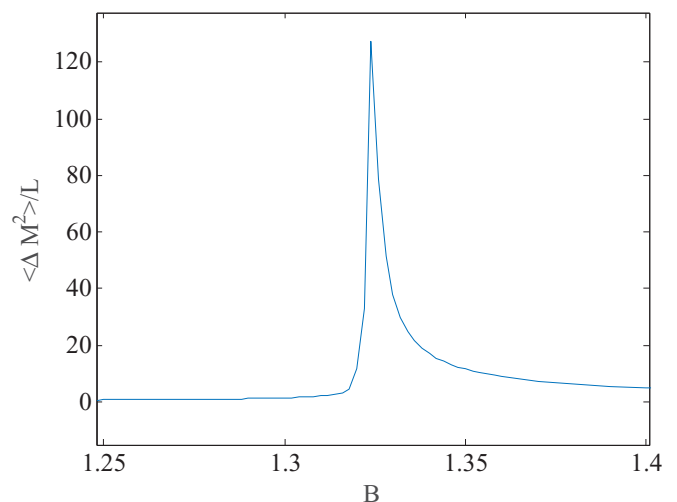

FIG. 12. (Color online) Per-site value of the second cumulant of the longitudinal magnetization $\frac{1}{L}\langle \Delta M_x^2 \rangle = \frac{1}{L}(\langle M_x^2 \rangle - \langle M_x \rangle^2)$, computed for the spin-1 Ising model. The cumulant is calculated for an infinite system directly.
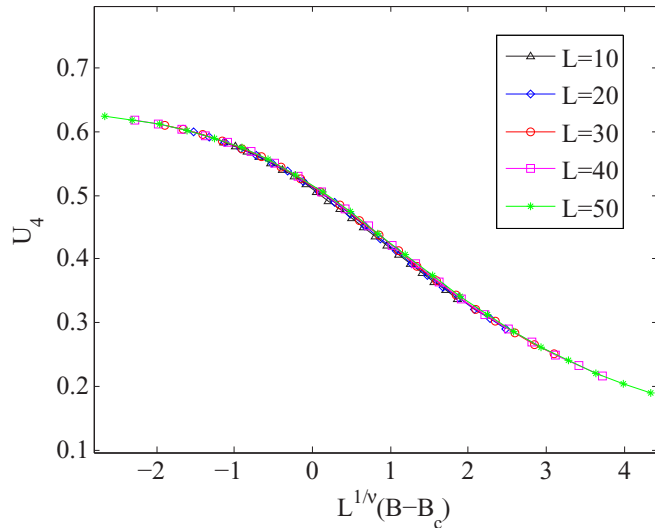
FIG. 13. (Color online) The Binder cumulants for the spin-1 transverse Ising model, plotted for a variety of system sizes as a function of $L^{1/\nu}(B - B_c)$, for the known values $\nu = 1$ and $B_c = 1.326$. The length independence of the curves allows this technique to be used as a means to estimate both ν and B_c

C. Spin-1 Ising model in crystal field

For another application we consider also a variation on the spin-1 Ising model, where the usual transverse field has been replaced by a quadratic, crystal field term, to give the following Hamiltonian:

$$H = -\sum_{j}^{L} S_j^x S_{j+1}^x + B\left(S_j^z\right)^2. \qquad (62)$$

This variation of the spin-1 model admits a mapping to the spin-1/2 case (cf. [76]), from which the critical point of $B_c = 2$ can be analytically obtained. To compare our method, we perform the same numerical calculations as above: first generating ground states at various finite lengths using MPS methods with a bond dimension of $\chi = 10$, and then computing the Binder cumulants. From the Binder curves (see Fig 14) we once again perform the BST extrapolation of the successive crossings to arrive at an estimate of $B_c = 1.999(1)$.

Direct examination of the second cumulant in the infinite system is also still a viable method for estimating the transition point. In this case, the location of the maximum gives $B_c = 1.996(1)$ (see Fig. 15). As before, it also remains possible to study the critical exponent ν and critical field value simultaneously by seeking to collapse the data to its universal behavior as a function of $L^{1/\nu}(B - B_c)$ (Fig. 16).

D. Spin-1/2 Ising model on a 2D lattice

Finally, as a proof-of-principle, we briefly demonstrate the application of these methods to a two-dimensional (2D) system: the spin-1/2 Ising model on a 2D square lattice. This system is described by the Hamiltonian

$$H = -\sum_{j,k} \sigma_{j,k}^x \left(\sigma_{j+1,k}^x + \sigma_{j,k+1}^x\right) + B S_{j,k}^z, \qquad (63)$$
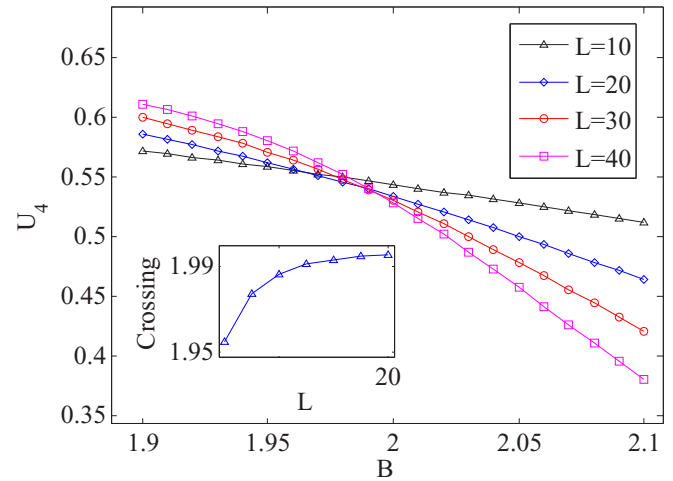

FIG. 14. (Color online) A Binder cumulant study of the spin-1 transverse Ising model with crystal field. As above, the cumulants are computed for different system sizes across a range of values for the transverse field B (some intermediate system sizes have been suppressed for clarity of the figure). Crossing points are interpolated for successive pairs of curves, i.e., $L = 10$ and $L = 15$, and the BST algorithm is used to extrapolate these values to the infinite limit, which gives $B_c = 1.999(1)$.

where the subscripts are understood to terminate at the boundary of the system. As discussed above, the study of two-dimensional systems with tensor network states is considerably more involved than the study of one-dimensional systems. More elaborate methods must be undertaken to numerically approximate the ground states, and elaborate approximation schemes must be used in order to calculate expectation values, which are otherwise prohibitively costly in time and memory. A detailed and high-precision study of the critical point of this model is therefore beyond the scope of this paper (see instead [20]). Nevertheless, we include the
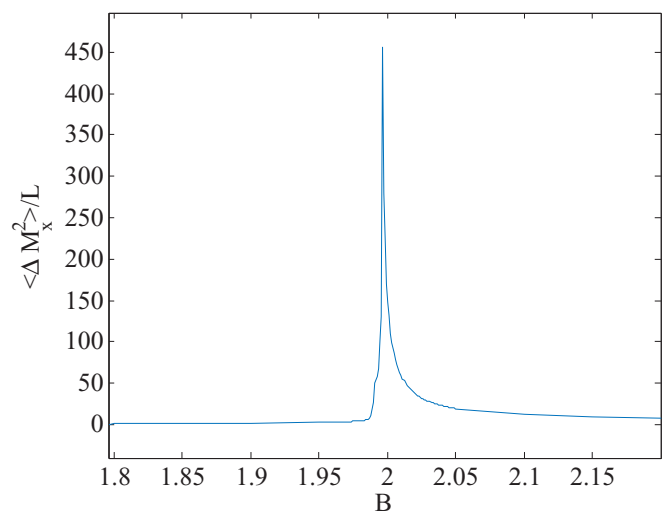

FIG. 15. (Color online) Per-site value of the second cumulant of the longitudinal magnetization $\frac{1}{L}\langle \Delta M_x^2 \rangle = \frac{1}{L}(\langle M_x^2 \rangle - \langle M_x \rangle^2)$, computed for the spin-1 Ising model with a transverse crystal field. The cumulant is calculated for an infinite system.
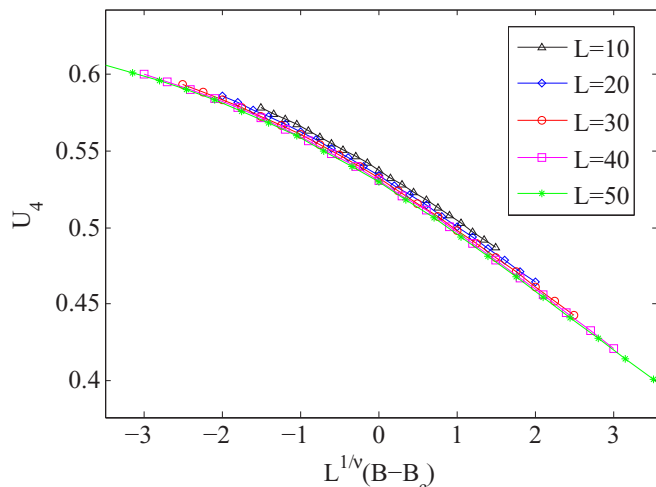
FIG. 16. (Color online) The Binder cumulants for the spin-1 Ising model with crystal field, plotted for a variety of system sizes as a function of $L^{1/\nu}(B - B_c)$ for the known values $\nu = 1$ and $B_c = 2$. As expected, for these values the curves are seen to collapse to a functional form essentially independent of the length scale.

following rough estimation in order to demonstrate how easily the moment-generating function method can be generalized to higher-dimensional states, as well as to underscore the utility of Binder cumulant techniques even for data calculated relatively cheaply.

To this end, we generate approximate ground states for the model, using a simple method of local updates (a 2D generalization of TEBD) and the smallest nontrivial bond dimension $\chi = 2$. To check the behavior, we consider the order parameter

$$M = \sum_{j,k} \sigma_{j,k}^x. \tag{64}$$

Then, as in the one-dimensional case, we compute the Binder cumulant of the order parameter across a range of applied fields, for systems of size $L \times L$ up to $L = 12$, and observe the crossings (Fig. 17). The largest crossing we are able to compute, between $L = 10$ and $L = 12$, occurs at $B = 3.11(1)$, which is already reasonably accurate compared to the accepted value of $B_c = 3.044$, as calculated by quantum Monte Carlo [77]. A BST extrapolation of the data gives $B_c = 3.3(2)$, a crude estimation but with relatively large error bars, owing largely to the fact that only three crossing values have been used in the extrapolation. We note also that, unlike the case of one-dimensional systems, the Binder crossings for two-dimensional systems are not necessarily converging monotonically and hence may not necessarily admit an easy extrapolation. Instead, greater precision could likely be obtained through the use of more sophisticated two-dimensional methods (or additional computational resources) to study the crossings for slightly larger system sizes.

VI. SUMMARY

In this paper we have presented a method for efficiently calculating the higher-order moments and cumulants of general
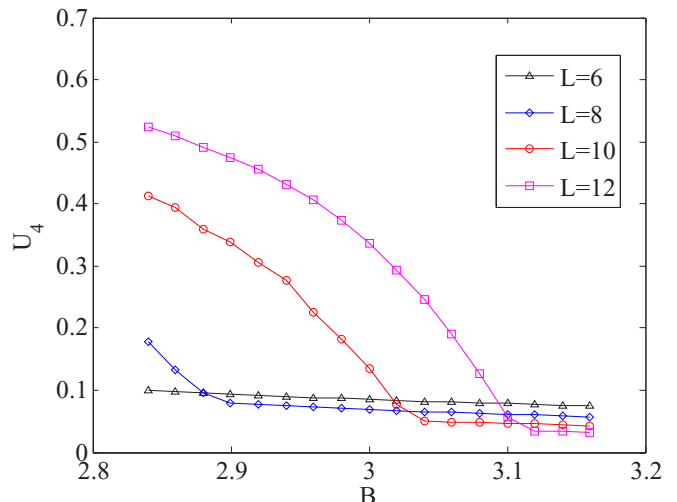

FIG. 17. (Color online) A rough Binder cumulant study of the spin-1/2 transverse Ising model on a square lattice, using a local-update numerical algorithm with bond dimension $\chi = 2$. As in the one-dimensional case, the cumulants are computed for different system sizes across a range of values for the transverse field B. The largest crossing point, between $L = 10$ and $L = 12$, is at $B = 3.11(1)$. Extrapolating to the infinite limit gives $B_c = 3.3(2)$, though this cannot be done with high reliability on such a limited data set (see text).

operators on systems represented by tensor network states. For finite systems, this capability has a variety of applications in the search for phase transitions in quantum systems. Chief among these is the calculation of the celebrated "Binder cumulant," which provides a powerful tool for not only detecting phase transitions, but determining their location to a high degree of accuracy using only relatively small finite systems to probe the infinite limit. The finite size scaling of the Binder cumulant also provides an estimate of the critical exponent of the correlation length. Although the second cumulants of Hamiltonians have been considered in the context of matrix product states, to our knowledge, critical point detection techniques based on the Binder cumulant (or cumulants in general) have not generally been put to use in studies based on tensor networks, despite being widely applied to classical systems and quantum Monte Carlo studies. It is our hope that the methods presented in this paper will allow them to be embraced by the tensor network community as well.

In the case of infinite systems, we also present a method to calculate the per-site limits of the cumulants efficiently as well. The higher cumulants of an order parameter often show sharp behavior at the critical points, which in many cases allows for easier detect than the changes in the order parameter itself. In particular, we show how singularities in the second cumulant can produce a relatively precise (computationally cheap) estimation of the location of the transition. All the techniques (finite and infinite) are demonstrated in the context of the transverse Ising model, the spin-1 transverse Ising model, the Ising model in a crystal field, and could easily be applied to other models. We also demonstrate a useful application of the second cumulant of the energy. This quantity, which we calculate for both finite and infinite systems,

provides a useful sufficient condition to determine when a numerical ground-state estimation algorithm has converged. As we demonstrate in the context of the Ising model, it can identify convergence up to a very high level of precision.

Finally, we present a proof-of-principle demonstration of the methods as applied to the transverse Ising model on a square lattice. Our result demonstrates that the methods for computing moments and cumulants are easily generalized to states in two dimensions or higher. Precise calculation of the moments and cumulants of a such a system may be more difficult, since state preparation and the process of computing expectation values are themselves much more complicated in higher dimensions. However, the central idea of our method on its own remains just as straightforward as in one dimension.

During the preparation of this work, it was brought to our attention that a technique based on matrix product operators (MPOs) has also been suggested as an alternative method for evaluating cumulants and moments, particularly in the context of the second cumulant of the energy and its use as a convergence check. We believe both methods have complimentary strengths and weaknesses, depending on the context in which they are applied.

APPENDIX A: EXTRAPOLATION WITH THE BST ALGORITHM

We now briefly overview the Bulirsch-Stoer extrapolation scheme, commonly referred to as the "BST" algorithm (the meaning of the "T" in this acronym has evidently been lost to time). This method was introduced in [70] in the context of differential equations, but has been widely adopted as an extrapolation scheme whenever one seeks to project a sequence of data with unknown functional form to its infinite limit. In particular it has become a useful tool for finite-size scaling techniques, and was studied extensively in the context of lattice models in [71].

The BST algorithm assumes that we are attempting to extrapolate to a limiting value for an infinite system, which is subject to power-law corrections when approximated by a finite size. For example, suppose we have a sequence of critical field values which are approximants to the true value of the critical field in the infinite system: $\{B(L_1), B(L_2), \ldots, B(L_N)\}$, which approach $B_\infty \equiv B(L \to \infty)$. The BST algorithm applies when, for each estimate $B(L)$, $B_\infty - B(L) = P(L)$ for some fixed (but unknown) polynomial P. This pattern of power-law corrections has generally been found to be true in the case of Binder cumulants [78,79].

The technique works by taking the initial sequence $\{\alpha_1^{(0)}, \alpha_2^{(0)}, \ldots, \alpha_N^{(0)}\}$ and using it to construct a new sequence, $\{\alpha_1^{(1)}, \alpha_2^{(1)}, \ldots, \alpha_{k-1}^{(1)}\}$ whose convergence towards the infinite limit has been accelerated, so that $\alpha_{N-1}^{(1)}$ is in fact a better estimate than $\alpha_N^{(0)}$. For clarity, note that we are using parenthetical superscripts to label the sequence, and subscripts to enumerate the terms within a sequence.

The terms in this new sequence are defined as follows:

$$\alpha_k^{(j+1)} \equiv \alpha_{k+1}^{(j)} + \frac{\alpha_{k+1}^{(j)} - \alpha_k^{(j)}}{\left(\frac{L_{k+1}}{L_k}\right)^\omega \left(1 - \frac{\alpha_{k+1}^{(j)} - \alpha_k^{(j)}}{\alpha_{k+1}^{(j)} - \alpha_{k+1}^{(j-1)}}\right) - 1}. \quad \text{(A1)}$$

Note that, since the denominator of Eq. (A1) makes reference to the sequence $\alpha^{(j-1)}$, it is necessary to define the sequence $\{\alpha^{(-1)}\}$ to handle the initial step of the algorithm in which $j = 0$. To that end, one simply takes $\alpha_k^{(-1)} = 0$ for all k.

This procedure can then be repeated, taking the sequences $\alpha^{(0)}$ and $\alpha^{(1)}$ as the inputs to generate $\alpha^{(2)}$, and so on. This iteration can be done at most $N - 1$ times, at which point the resulting sequence $\alpha^{(N-1)}$ contains only one term. This term is the BST algorithm's best estimation of the infinite limit of the original sequence.

The parameter "ω," which appears as the exponent on the length scales, is a free parameter in the algorithm. The value of ω which gives the best convergence will depend on the form of the power law corrections in the original sequence, which is generally unknown. Hence, in practice, a range of parameters must be considered, selecting the one which best optimizes the convergence. To this end, note that a rough estimate of the "precision" of the sequences can be made by

$$\Delta_k^{(j+1)} = 2\left|\alpha_k^{(j)} - \alpha_{k+1}^{(j)}\right|. \quad \text{(A2)}$$

This value should be decreasing with each iteration if the procedure is correctly accelerating the convergence of each new sequence. The final value of this estimator $\Delta_{\text{final}} = \Delta_1^{(N-1)}$ gives a convenient way to fix the free parameter ω: we simply repeat the algorithm while varying ω, and choose the one which minimizes Δ_{final}. In practice it has often been observed [80] that the dependence of the estimations on ω is rather weak, with large ranges of values giving comparable results. In other words, it is often more important simply to avoid a "bad" value of ω than to try to find its absolute "best" value. In our work we have used as our procedure a sweep over the range $\omega \in (0,2]$, testing with steps of size 0.1. We also require that our extrapolation be "stable" under small variations in ω.

We note that the value of Δ_{final} cannot be used as a complete measure of the error in a final estimation. It is a measure of the internal consistency and the precision of the acceleration in the BST algorithm, but cannot contain any information about whether the algorithm has captured the "true" functional form of the finite-size corrections. Additionally, it does not reflect the propagation of errors on the data which are being extrapolated. A very small and stable value of Δ_{final} indicates that the algorithm is extrapolating the data to the best of its capability given its assumptions and the finite number of input points. It does not necessarily indicate that the result is extremely precise.

In this paper, to estimate the error in a BST extrapolation, we start with the uncertainties of the input points, and essentially determine the propagated uncertainty empirically. In our case, the input points are the crossings of Binder cumulant curves. The crossings are computed by linearly extrapolating between data points, so we generously assume an uncertainty of one half the step size between points. The error is then estimated by considering a "worst-case scenario" in which the first few crossing points are perturbed downward by this amount, and the later points perturbed upwards. We run these perturbed points through the BST algorithm and observe the effect on the resulting extrapolation. The size of this effect is taken to be a rough upper bound on the total uncertainty.

APPENDIX B: COMMENTS ON CONVERGENCE SCHEMES

When finding a ground state in an algorithm such as TEBD or iTEBD (and indeed also DMRG and iDMRG), the function of each iteration is to rotate the current state vector closer to an eigenstate with a lower energy. It is therefore quite natural and typical to define the halting condition of such a scheme by reference to the energy of the current state. In the simplest instance, for example, one might set a convergence threshold of ϵ and then, after each step i of the algorithm, compute the resulting change in the state's energy, $\Delta E = E_i - E_{i+1}$, with the understanding that the iteration will halt if ever $\Delta E \leqslant \epsilon$. A more sophisticated version of the same idea would be to compute the standard deviation for the N most recent steps instead, to make the process less susceptible to small numerical fluctuations.

Depending upon the quantities of interest, however, and the desired levels of precision, this type of convergence framework may not be optimal. For one thing, as discussed above, one must be careful of instance of "metastability" in the energy values, which may produce periods of time during the algorithm where the energy is changing very slowly and yet is still far from the true ground-state value. This, in part, is the reason why a typical iTEBD convergence scheme consists of a set of sequential iterations, each time with a smaller imaginary time step $\delta\tau$ [16], since it is quite common to see the energy values experience a greater change after the step size is reduced (see Fig. 18). Even with decreasing step sizes, however, it is possible for the changes in the energy to stall into metastability. It is here that checking the second cumulant of the energy, as discussed in the main text, can be advantageous, since a small value of this cumulant rules out the possibility that the current energy is far from that of the true ground state.

It is also important to consider, however, that not all quantities will converge at the same rate. In Fig. 19, for example, we show the results of a study of the spin-1 Heisenberg model with a relatively large transverse field. We consider the observables E and $\langle S_x \rangle$, as well as the entanglement entropy S (note that the state must be put in canonical form before the exact entanglement entropy can be directly calculated from the tensors). After every 100 steps of evolution we compute for each quantity the coefficient of variation [the standard deviation divided by the mean, $\sigma(X)/\bar{X}$] over a sample of the previous 300 steps, to determine how much the quantities are currently being "changed" by the algorithm. Since these
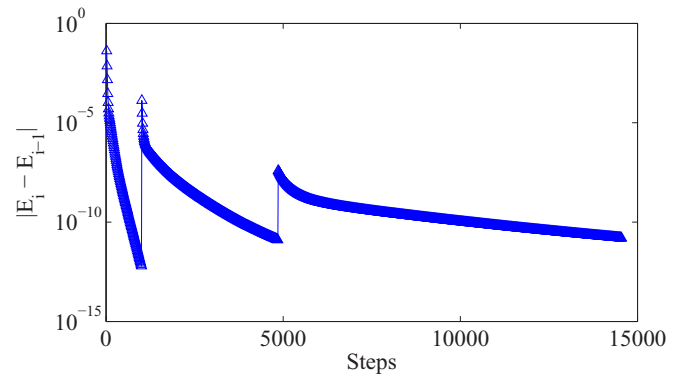

FIG. 18. (Color online) A test of the energy convergence behavior during the iTBED algorithm, as applied to the spin-1 Heisenberg model with a transverse field. After each step, the difference in energy compared to the previous step is plotted. The segments separated by sharp transitions indicate the different stages of the algorithm, with the size of the imaginary time step $\delta\tau$ decreased for each subsequent stage. Because the changes in the energy can become so small during the course of any individual stage, one might erroneously believe that convergence had been achieved, but moving to a smaller time step reveals that this is not so.

coefficients of variation are dimensionless, different quantities can be compared directly in terms of their convergence. As can be seen in the figure, the number of steps sufficient to achieve a stable value of the energy out to a high precision is not at all sufficient to achieve the same accuracy in the other quantities. A study which focused on the magnetization of ground states, for example, might be better off using the stability of the magnetizations themselves, rather than the energies, to certify convergence.
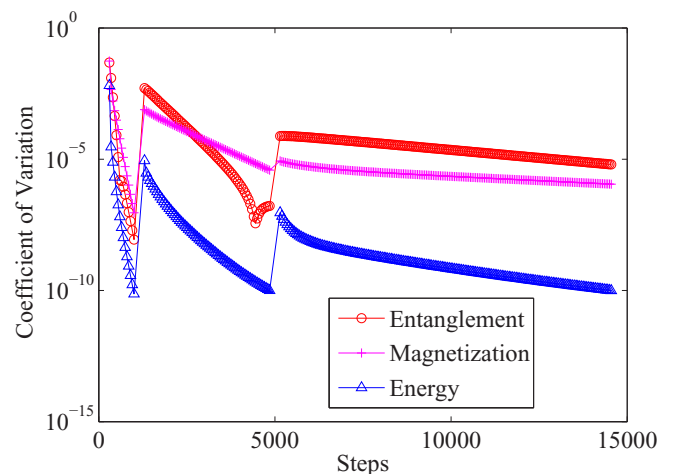

FIG. 19. (Color online) A test of the convergence behavior of E, S, and $\langle S_x \rangle$ during the iTBED algorithm, as applied to the spin-1 Heisenberg model with a transverse field. Every 100 steps, the previous 300 steps are sampled and the standard deviation is computed, and then divided by the mean. The resulting coefficient of variation is plotted in the figure above. The curve for the energy is consistently much lower, indicating that this quantity converges much more quickly than the others.
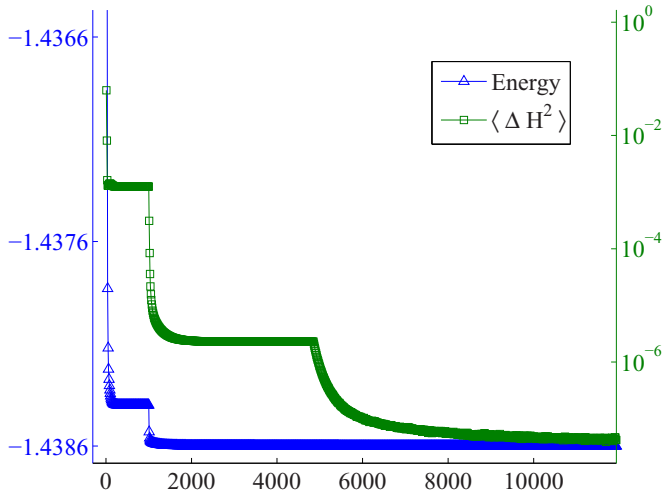
FIG. 20. (Color online) A test comparing the convergence behavior of the energy (left axis) and $\langle \Delta H^2 \rangle$ (right axis) during the iTBED algorithm as applied to the spin-1 Heisenberg model with a transverse field. The underlying data are the same as Fig. 19. Though the energy has converged to the level of one part in 10^5 after approximately 1000 steps, the energy cumulant takes much longer to reach a similar level of precision. In part, this is because the cumulant requires evolution with smaller time steps, in order to avoid plateaus and fully converge.

Unfortunately, it does not seem to be the the case that any one particular quantity is universally more difficult to converge than the others; instead, the relative convergence rates tend to depend on the Hamiltonian being studied. Therefore, when precision is required in more than one observable, one option is to to perform a small convergence study first, to determine which quantity will produce the strictest convergence across the board. We do note, however, that since algorithms like iTEBD operate directly on the energy of the state, the energy is likely to be among the *fastest* quantities to converge, making it a poor choice for a convergence criterion if other quantities are desired as well. Interestingly, however, in our experience the same is not true for the second cumulant of the energy $\langle \Delta H^2 \rangle$, which is often (though likely not always) slower to converge than the energy (see for example Fig. 20) and which frequently seems not to converge until other observables like the magnetizations have converged as well. Since this quantity can also be efficiently computed using an MPO or a cumulant generating function, it may be a good convergence criterion for a variety of models.

Finally, having chosen which quantity or quantities to monitor for convergence, it remains to decide which sequence of imaginary time steps should be used. Here, too, the optimal choice may depend on the model being considered, and perhaps also the bond dimension of the MPS representing it. In general, however, the precise set of steps sizes chosen is likely less important than it may initially appear. To see this, consider the possible consequences of a poor choice of convergence scheme. Either the evolution may "waste time" with too large a time step if it winds up in a plateau, or, if the time steps become too small too quickly, it may fail to change the state enough to ever approach the true ground state. Both of these problems are demonstrated in Fig. 21, in which
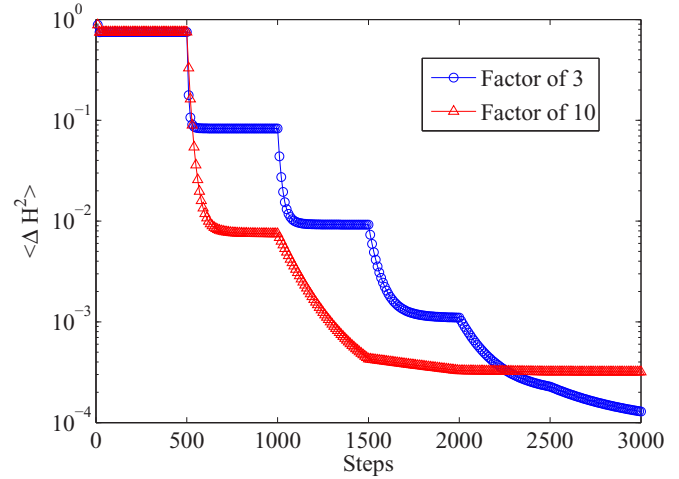

FIG. 21. (Color online) The transverse Ising model on a chain of $L = 30$ is studied by a basic TEBD algorithm. The initial imaginary time step is $\delta\tau = 0.1$. After every 500 steps of evolution, the step size is decreased either by a factor of 3 (triangles) or a factor of 10 (circles). In the early stages (steps 500–1500) the low-factor scheme is outperformed, because it spends more time stuck in local plateaus. In the later stages, however, the large-factor scheme falls behind. The evolution between steps 1000 and 1500 is truncated too early, before it had locally converged, and the subsequent time steps are too small to cause a meaningful change in the state.

we compare two different convergence schemes. Both begin with an initial step size of $\delta\tau = 0.1$, and in each case, after a fixed number of steps, the step size is decreased by a constant reduction factor. At early stages, it seems to be a problem to
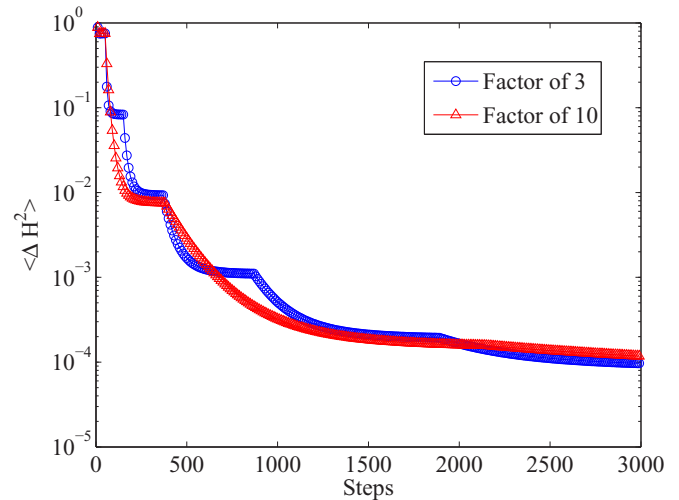

FIG. 22. (Color online) The transverse Ising model on a chain of $L = 30$ is studied by a basic TEBD algorithm. In contrast with the convergence scheme used in Fig. 21, here we use the coefficient of variation from the 30 most recent points to determine when to move on to a smaller step size. The result is that unnecessary plateaus are largely avoided, and no time step is terminated before it has finished producing a useful change in the state. As a result, the energy cumulant $\langle \Delta H^2 \rangle$ is always quite similar regardless of the step size. The final energy computed for the ground state differs by less than one part in 10^7.

use a small reduction factor, since the evolution quickly get stuck in a plateau at each stage. In the later stages, however, where the time steps are small, we see how a large reduction factor can cause problems as well, since the step size quickly becomes too small to produce any meaningful change.

Both of these problems, however, can be viewed not as the result of improper step sizing, but rather as the result of the fact that we used a fixed number of steps at each stage. If we employ a convergence check of the kind described above (for example, computing the coefficient of variation of the 30 most recent points) and move on to the next size time step whenever the data have locally converged, the differences in the two algorithms largely disappear (see Fig. 22). Time is no longer wasted in plateaus, since a new step size is triggered whenever the data have stopped changing significantly. On the other hand, the small time steps are never employed prematurely because the algorithm does not move on to a smaller step

until the current step has outlived its usefulness. Hence, by requiring local convergence before changing the size of the time step, the differences between different sets of step sizes can be largely "smoothed out." Although there may be small advantages to a particular scheme in the context of a specific Hamiltonian, in general any set of step sizes that contains some intermediate-sized points can be successful (in practice, this likely means reduction factors between roughly 2 and 10).

Finally, we note that of course, in any convergence scheme, it must be remembered that the calculations involved in a convergence check can themselves contribute substantially to an algorithm's run time, and one must take care to balance this against the desire to run in an optimal number of steps. In practice, the most important feature of an algorithm is likely the amount of real-world time taken to reach the desired level of precision, and the sophistication and frequency of the convergence checks can be relaxed if they are at odds with this goal.

[1] J. I. Cirac and F. Verstraete, Renormalization and tensor product states in spin chains and lattices, J. Phys. A: Math. Theor. **42**, 504004 (2009).

[2] F. Verstraete, V. Murg, and J. I. Cirac, Matrix product states, projected entangled pair states, and variational renormalization group methods for quantum spin systems, Adv. Phys. **57**, 143 (2008).

[3] D. Perez-Garcia, F. Verstraete, M. M. Wolf, and J. I. Cirac, Matrix product state representations, Quantum Inf. Comput. **7**, 401 (2007).

[4] R. Orús, A practical introduction to tensor networks: Matrix product states and projected entangled pair states, Ann. Phys. **349**, 117 (2014).

[5] M. Fannes, B. Nachtergaele, and R. F. Werner, Finitely correlated states on quantum spin chains, Commun. Math. Phys. **144**, 443 (1992).

[6] A. Klümper, A. Schadschneider, and J. Zittartz, Equivalence and solution of anisotropic spin-1 models and generalized TJ fermion models in one dimension, J. Phys. A: Math. Gen. **24**, L955 (1991).

[7] A. Klümper, A. Schadschneider, and J. Zittartz, Matrix product ground states for one-dimensional spin-1 quantum antiferromagnets, Europhys. Lett. **24**, 293 (1993).

[8] S. R. White, Density Matrix Formulation for Quantum Renormalization Groups, Phys. Rev. Lett. **69**, 2863 (1992).

[9] S. R White, Density-matrix algorithms for quantum renormalization groups, Phys. Rev. B **48**, 10345 (1993).

[10] S. Östlund and S. Rommer, Thermodynamic Limit of Density Matrix Renormalization, Phys. Rev. Lett. **75**, 3537 (1995).

[11] T. Nishino, T. Hikihara, K. Okunishi, and Y. Hieida, Density matrix renormalization group: Introduction from a variational point of view, Int. J. Mod. Phys. B **13**, 1 (1999).

[12] U. Schollwöck, The density-matrix renormalization group, Rev. Mod. Phys. **77**, 259 (2005).

[13] U. Schollwöck, The density-matrix renormalization group in the age of matrix product states, Ann. Phys. **326**, 96 (2011).

[14] G. Vidal, Efficient Simulation of One-Dimensional Quantum Many-Body Systems, Phys. Rev. Lett. **93**, 040502 (2004).

[15] G. M. Crosswhite, A. C. Doherty, and G. Vidal, Applying matrix product operators to model systems with long-range interactions, Phys. Rev. B **78**, 035116 (2008).

[16] G. Vidal, Classical Simulation of Infinite-Size Quantum Lattice Systems in One Spatial Dimension, Phys. Rev. Lett. **98**, 070201 (2007).

[17] G. Vidal, Entanglement Renormalization, Phys. Rev. Lett. **99**, 220405 (2007).

[18] G. Evenbly and G. Vidal, Class of Highly Entangled Many-Body States That Can Be Efficiently Simulated, Phys. Rev. Lett. **112**, 240502 (2014).

[19] F. Verstraete and J. I. Cirac, Renormalization algorithms for quantum-many body systems in two and higher dimensions, arXiv:cond-mat/0407066.

[20] J. Jordan, R. Orús, G. Vidal, F. Verstraete, and J. I. Cirac, Classical Simulation of Infinite-Size Quantum Lattice Systems in Two Spatial Dimensions, Phys. Rev. Lett. **101**, 250602 (2008).

[21] D. Perez-Garcia, F. Verstraete, M. M. Wolf, and J. I. Cirac, PEPs as unique ground states of local Hamiltonians, Quantum Inf. Comput. **8**, 650 (2008).

[22] N. Schuch, J. I. Cirac, and D. Pérez-García, PEPs as ground states: Degeneracy and topology, Ann. Phys. **325**, 2153 (2010).

[23] L. Tagliacozzo, G. Evenbly, and G. Vidal, Simulation of two-dimensional quantum systems using a tree tensor network that exploits the entropic area law, Phys. Rev. B **80**, 235127 (2009).

[24] Y.-Y. Shi, L.-M. Duan, and G. Vidal, Classical simulation of quantum many-body systems with a tree tensor network, Phys. Rev. A **74**, 022320 (2006).

[25] M. Cozzini, R. Ionicioiu, and P. Zanardi, Quantum fidelity and quantum phase transitions in matrix product states, Phys. Rev. B **76**, 104420 (2007).

[26] M. Asoudeh, V. Karimipour, and A. Sadrolashrafi, Quantum phase transitions and matrix product states in spin ladders, Phys. Rev. B **75**, 224427 (2007).

[27] F. Heidrich-Meisner, I. P. McCulloch, and A. K. Kolezhuk, Phase diagram of an anisotropic frustrated ferromagnetic spin-1/2 chain in a magnetic field: A density matrix renormalization group study, Phys. Rev. B **80**, 144417 (2009).

[28] J. A Kjäll, M. P. Zaletel, R. S. K. Mong, J. H. Bardarson, and F. Pollmann, Phase diagram of the anisotropic spin-2 xxz model: Infinite-system density matrix renormalization group study, Phys. Rev. B **87**, 235106 (2013).

[29] K. Binder, Finite size scaling analysis of Ising model block distribution functions, Z. Phys. B Condens. Matter, **43**, 119 (1981).

[30] H. Hong, H. Park, and L. H. Tang, Anomalous binder cumulant and lack of self-averageness in systems with quenched disorder, J. Korean Phys. Soc. **49**, L1885 (2006).

[31] R. N. Bhatt and A. P. Young, Search for a Transition in the Three-Dimensional j Ising Spin-Glass, Phys. Rev. Lett. **54**, 924 (1985).

[32] N. Kawashima and A. P. Young, Phase transition in the three-dimensional $\pm j$ Ising spin glass, Phys. Rev. B **53**, R484 (1996).

[33] K. Kremer and K. Binder, Monte Carlo simulation of lattice models for macromolecules, Comput. Phys. Rep. **7**, 259 (1988).

[34] W. Selke, Critical binder cumulant of two-dimensional Ising models, Eur. Phys. J. B **51**, 223 (2006).

[35] W. Selke, The critical binder cumulant for isotropic Ising models on square and triangular lattices, J. Stat. Mech: Theory Exp. (2007) P04008.

[36] W. Selke and L. N. Shchur, Critical binder cumulant in a two-dimensional anisotropic Ising model with competing interactions, Phys. Rev. E **80**, 042104 (2009).

[37] A. W. Sandvik, Evidence for Deconfined Quantum Criticality in a Two-Dimensional Heisenberg Model with Four-Spin Interactions, Phys. Rev. Lett. **98**, 227202 (2007).

[38] H. P. Ying, H. J. Luo, L. Schülke, and B. Zheng, Dynamic Monte Carlo study of the two-dimensional quantum xy model, Mod. Phys. Lett. B **12**, 1237 (1998).

[39] J.-K. Kim and A. Patrascioiu, Critical Behavior of the Specific Heat in the Two Dimensional Site Diluted Ising system, Phys. Rev. Lett. **72**, 2785 (1994).

[40] H. G. Ballesteros, A. Cruz, L. A. Fernández, V. Martín-Mayor, J. Pech, J. J. Ruiz-Lorenzo, A. Tarancón, P. Téllez, C. L. Ullod, and C. Ungil, Critical behavior of the three-dimensional Ising spin glass, Phys. Rev. B **62**, 14237 (2000).

[41] G. Grégoire and H. Chaté, Onset of Collective and Cohesive Motion, Phys. Rev. Lett. **92**, 025702 (2004).

[42] A. Weichselbaum, F. Verstraete, U. Schollwöck, J. I. Cirac, and J. von Delft, Variational matrix-product-state approach to quantum impurity models, Phys. Rev. B **80**, 165117 (2009).

[43] I. P. McCulloch, Infinite size density matrix renormalization group, revisited, arXiv:0804.2509.

[44] I. P. McCulloch, From density-matrix renormalization group to matrix product states, J. Stat. Mech: Theory Exp. (2007) P10014.

[45] F. Zhan, J. Sabbatini, M. J. Davis, and I. P. McCulloch, Miscible-immiscible quantum phase transition in coupled two-component Bose-Einstein condensates in one-dimensional optical lattices, Phys. Rev. A **90**, 023630 (2014).

[46] S. N. Saadatmand, B. J. Powell, and I. P. McCulloch, Phase diagram of the spin-1/2 triangular j_1-j_2 Heisenberg model on a 3-leg cylinder, arXiv:1503.07570 (2015).

[47] A. C. Davison, *Statistical Models*, Vol. 11 (Cambridge University Press, 2003).

[48] P. J. Smith, A recursive formulation of the old problem of obtaining moments from cumulants and vice versa, Am. Stat. **49**, 217 (1995).

[49] H. Yi, Quantum phase transition of the transverse-field quantum Ising model on scale-free networks, Phys. Rev. E **91**, 012146 (2015).

[50] R. Orús and G. Vidal, Infinite time-evolving block decimation algorithm beyond unitary evolution, Phys. Rev. B **78**, 155117 (2008).

[51] E. Lukacs, *Characteristic Functions* (Griffin, London, 1970), Vol. 5.

[52] W. Rudin, *Real and Complex Analysis* (Tata McGraw-Hill Education, New York, 1987).

[53] H. Neudecker, A note on Kronecker matrix products and matrix equation systems, SIAM J. Appl. Math. **17**, 603 (1969).

[54] B. Fornberg, Generation of finite difference formulas on arbitrarily spaced grids, Math. Comput. **51**, 699 (1988).

[55] F. Pollmann (private communication).

[56] V. Nebendahl (private communication).

[57] J. J. García-Ripoll, Time evolution of matrix product states, New J. Phys. **8**, 305 (2006).

[58] M. Suzuki, General theory of higher-order decomposition of exponential operators and symplectic integrators, Phys. Lett. A **165**, 387 (1992).

[59] I. P. Omelyan, I. M. Mryglod, and R. Folk, Optimized Forest-Ruth and Suzuki-like algorithms for integration of motion in many-body systems, Comput. Phys. Commun. **146**, 188 (2002).

[60] L. Michel and I. P. McCulloch, Schur forms of matrix product operators in the infinite limit, arXiv:1008.4667.

[61] H. N. Phien, G. Vidal, and I. P. McCulloch, Infinite boundary conditions for matrix product state calculations, Phys. Rev. B **86**, 245107 (2012).

[62] N. Schuch, M. M. Wolf, F. Verstraete, and J. I. Cirac, Computational Complexity of Projected Entangled Pair States, Phys. Rev. Lett. **98**, 140506 (2007).

[63] M. Levin and C. P. Nave, Tensor Renormalization Group Approach to Two-Dimensional Classical Lattice Models, Phys. Rev. Lett. **99**, 120601 (2007).

[64] R. Orús and G. Vidal, Simulation of two-dimensional quantum systems on an infinite lattice revisited: Corner transfer matrix for tensor contraction, Phys. Rev. B **80**, 094403 (2009).

[65] A. García-Sáez and J. I Latorre, Renormalization group contraction of tensor networks in three dimensions, Phys. Rev. B **87**, 085130 (2013).

[66] Z. Y. Xie, Jing Chen, M. P. Qin, J. W. Zhu, and L. P. Yang, and T. Xiang, Coarse-graining renormalization by higher-order singular value decomposition, Phys. Rev. B **86**, 045139 (2012).

[67] T. Nishino and K. Okunishi, Corner transfer matrix renormalization group method, J. Phys. Soc. Jpn. **65**, 891 (1996).

[68] P. Pfeuty, The one-dimensional Ising model with a transverse field, Ann. Phys. **57**, 79 (1970).

[69] S. Suzuki, J.-i. Inoue, and B. K. Chakrabarti, *Quantum Ising Phases and Transitions in Transverse Ising Models*, Vol. 862 (Springer, 2012), Chap. 2.

[70] R. Bulirsch and J. Stoer, Numerical treatment of ordinary differential equations by extrapolation methods, Numer. Math. **8**, 1 (1966).

[71] M. Henkel and G. Schutz, Finite-lattice extrapolation algorithms, J. Phys. A: Math. Gen. **21**, 2617 (1988).

[72] I. P. McCulloch (private communication).

[73] M. Henkel, *Conformal Invariance and Critical Phenomena* (Springer Science & Business Media, New York, 1999).

[74] T. Kaneyoshi, M. Jaščur, and I. P. Fittipaldi, Transverse Ising model with arbitrary spin, Phys. Rev. B **48**, 250 (1993).

[75] C. Liu, L. Wang, A. W. Sandvik, Y.-C. Su, and Y.-J. Kao, Symmetry breaking and criticality in tensor-product states, Phys. Rev. B **82**, 060410 (2010).

[76] J. Oitmaa and A. M. A. von Brasch, Spin-1 Ising model in a transverse crystal field, Phys. Rev. B **67**, 172402 (2003).

[77] H. W. J. Blöte and Y. Deng, Cluster Monte Carlo simulation of the transverse Ising model, Phys. Rev. E **66**, 066110 (2002).

[78] T. Jörg and H. G. Katzgraber, Universality and universal finite-size scaling functions in four-dimensional Ising spin glasses, Phys. Rev. B **77**, 214426 (2008).

[79] H. Chamati and D. M. Dantchev, Renormalization group treatment of the scaling properties of finite systems with subleading long-range interaction, Eur. Phys. J. B **26**, 89 (2002).

[80] N. A. Alves, J. R. Drugowich De Felicio, and U. H. E. Hansmann, Partition function zeros and leading-order scaling correction of the 3D Ising model from multicanonical simulations, J. Phys. A: Math. Gen. **33**, 7489 (2000).