The iTEBD algorithm beyond unitary evolution

R. Orús and G. Vidal

School of Physical Sciences, The University of Queensland, QLD 4072, Australia

The infinite *time-evolving block decimation* (iTEBD) algorithm [Phys. Rev. Lett. 98, 070201 (2007)] allows to simulate unitary evolution and to compute the ground state of one-dimensional quantum lattice systems in the thermodynamic limit. Here we extend the algorithm to tackle a much broader class of problems, namely the simulation of arbitrary one-dimensional evolution operators that can be expressed as a (translationally invariant) tensor network. Relatedly, we also address the problem of finding the dominant eigenvalue and eigenvector of a one-dimensional transfer matrix that can be expressed in the same way. New applications include the simulation, in the thermodynamic limit, of open (i.e. master equation) dynamics and thermal states in 1D quantum systems, as well as calculations with partition functions in 2D classical systems, on which we elaborate. The present extension of the algorithm also plays a prominent role in the infinite *projected entangled-pair states* (iPEPS) approach to infinite 2D quantum lattice systems.

I. INTRODUCTION

The development of numerical methods to explore the properties of strongly correlated many-body systems remains one of the most challenging problems in computational physics. In recent years, increasing attention has been paid to algorithms that express the state of the system as a tensor network. For instance, for quantum systems on a 1D lattice, a *matrix product state* (MPS) [1] represents the system's wave function in the *density matrix renormalization group* (DMRG) algorithm to compute ground states [2], and in the *time-evolving block decimation* (TEBD) algorithm to simulate time evolution [3]. Similarly, the *tensor product state* (TPS) [4] and the *projected entangled-pair state* (PEPS) [5] have been proposed to accomplish those tasks in 2D lattices, whereas the *multi-scale entanglement renormalization ansatz* (MERA) [6] is specially suited to describe systems at criticality or with topological order. Finally, tensor networks can also be used to encode and manipulate the partition function of 2D classical lattice systems [7, 8, 9].

The computational cost of a simulation using tensor network algorithms is roughly proportional to the size of the lattice. However, when the system is invariant under translations, this cost can be made independent of the system's size. The infinite TEBD (iTEBD) algorithm [7] exploits this fact to simulate unitary evolution and compute the ground state of a 1D quantum system in the limit of an infinite lattice. The key idea is to encode the wave function in an infinite MPS (iMPS) made of a small number of tensors that are repeated indefinitely and, importantly, to maintain the iMPS in its canonical form during the whole simulation. As a result, bulk properties of 1D quantum systems are computed directly in the thermodynamic limit, circumventing more costly and less accurate approaches based on finite-size scaling. Other algorithms, such as the *power wave function renormalization group* (PWFRG) [10] or infinite DMRG (iDMRG) [11] also compute the ground state of infinite systems.

A major limitation of the iTEBD algorithm is that it can only address unitary evolution [as explained in Sect. III, the computation of ground states with imaginary time evolution is a lucky exception]. Thus, the simulation of more general types of evolution, such as master equation evolution in a dissipative system or imaginary time evolution to compute thermal states, is still restricted to finite systems [13, 14]. The reason lies in the fact that only unitary evolution preserves the canonical form of the iMPS. The latter is essential in order to keep truncation errors small during the simulation. Indeed, in the absence of the canonical form, truncation errors accumulate unnecessarily fast and ruin the simulation in places where an efficient iMPS description would otherwise still be feasible.

In this paper we explain how to overcome such shortcoming. First we describe how to compute the canonical form of an iMPS. Then we present an extension of the iTEBD algorithm that is able to simulate a much wider class of evolution. Namely, it simulates the action on an iMPS of any transformation that can be expressed as a translationally invariant tensor network. This includes, as particular cases, evolution in imaginary time or according to a master equation. We also explain how to use the algorithm to compute the dominant eigenvalue and eigenvector [15] of any one-dimensional transfer matrix that decomposes as a translationally invariant tensor network. As an application of this, we explain how to extract correlators and local observables from the partition function of a 2D classical system. Finally, the extended version presented in this work plays a prominent role in the *infinite* PEPS (iPEPS) algorithm to simulate evolution and compute ground states in infinite 2D quantum lattice systems [16], as well as in certain implementations of MERA algorithms [6].

We emphasize that other algorithms can be used to address infinite systems, and that they are also based on or related to computing the dominant eigenvalue and eigenvector of a one-dimensional transfer matrix. This is the case, for instance, of the PWFRG and iDMRG algorithms [10, 11] to compute ground states in 1D quantum systems and the transfer matrix renormalization group (TMRG) algorithm [7] to evaluate partion functions in 2D classical systems. However, iTEBD differs from them at its core in two important aspects: first, TMRG, PWFRG and iDMRG are variational methods, while iTEBD amounts to a power method; second, whereas TMRG, PWFRG and iDMRG converge towards an infinite system by

adding sites to a finite lattice, in iTEBD the system is infinite from the onset. We notice that Ref. [11], which contains a useful comparative study, highlights that the iTEBD is significantly more accurate than other proposals in determining ground states. Nevertheless, all these methods are of comparable interest.

The rest of the paper is organized as follows. Sect. II explains how to obtain the canonical form of an iMPS by orthonormalizing all its bond indices. Sect. III presents the generalization the iTEBD algorithm to account for non-unitary evolution. Sect. IV discusses an application of the algorithm to 2D classical lattice models and Sect. V contains some conclusions. Finally, the appendix presents a detailed description of how to implement the algorithm for very specific forms of the evolution operator.

II. CANONICAL FORM OF AN INFINITE MPS

We consider an infinite 1D lattice, where each site is labeled by an integer $r \in \mathbb{Z}$ and described by a Hilbert space $\mathbb{C}^d$ of finite dimension d. The lattice is in a pure state $|\Psi\rangle \in \bigotimes_{r \in \mathbb{Z}} \mathbb{C}^d$ that is invariant under translations by n sites (and multiples thereof). Following Ref. [12], we represent $|\Psi\rangle$ using an iMPS, which in the simplest case ($n = 1$) consists of a pair of tensors $\{\Gamma, \lambda\}$, see Figs. (1.i)-(1.ii). Here Γ is made of complex coefficients $\Gamma^i_{\alpha\beta}$, with two *bond* indices α and β ($\alpha, \beta = 1, \cdots, \chi$) and one *physical* index i ($i = 1, \cdots, d$) that labels an orthonormal basis in $\mathbb{C}^d$, whereas λ is a diagonal matrix with non-negative diagonal elements λ_α. The integer χ is known as the rank of the iMPS.

We say that an iMPS $\{\Gamma, \lambda\}$ is in its *canonical form* [3, 17] when, on each bond, index α is related to the Schmidt decomposition of $|\Psi\rangle$,

$$|\Psi\rangle = \sum_{\alpha=1}^{\chi} \lambda_\alpha |\Phi^L_\alpha\rangle \otimes |\Phi^R_\alpha\rangle \, , \tag{1}$$

that is, when the diagonal matrix λ contains the decreasingly ordered Schmidt coefficients ($\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_\chi \geq 0$) and α labels the Schmidt vectors, which form orthonormal sets, $\langle\Phi^L_\alpha|\Phi^L_{\alpha'}\rangle = \langle\Phi^R_\alpha|\Phi^R_{\alpha'}\rangle = \delta_{\alpha\alpha'}$. In terms of the matrices R and L, defined as

$$R_{(\alpha\alpha'),(\beta\beta')} \equiv \sum_{i=1}^{d} \left(\Gamma^i_{\alpha\beta}\lambda_\beta\right)\left(\Gamma^i_{\alpha'\beta'}\lambda_{\beta'}\right)^* \tag{2}$$

$$L_{(\alpha\alpha'),(\beta\beta')} \equiv \sum_{i=1}^{d} \left(\lambda_\alpha\Gamma^i_{\alpha\beta}\right)\left(\lambda_{\alpha'}\Gamma^i_{\alpha'\beta'}\right)^* \tag{3}$$

the canonical form corresponds to the conditions

$$\sum_{\beta,\beta'} R_{(\alpha\alpha'),(\beta\beta')}\delta_{\beta\beta'} = \eta\delta_{\alpha\alpha'} \tag{4}$$

$$\sum_{\alpha,\alpha'} \delta_{\alpha\alpha'} L_{(\alpha\alpha'),(\beta\beta')} = \eta\delta_{\beta\beta'} \, , \tag{5}$$
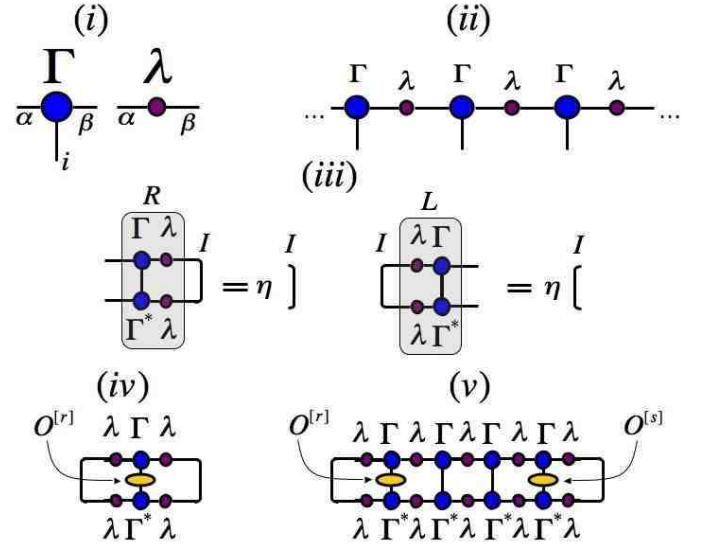

Figure 1: (color online) (i) Diagrammatic representation of the tensors Γ and λ that form an iMPS for a pure state $|\Psi\rangle$ of a translationally invariant chain. (ii) The iMPS is actually an infinite one-dimensional tensor network consisting of alternating copies of the tensors Γ and λ. In the diagram, a leg shared by two tensors corresponds to an index over which there is an implicit sum. (iii) Diagrammatic representation of the conditions in Eqs. (4)-(5), which are fulfilled if and only if the iMPS $\{\Gamma, \lambda\}$ is in its canonical form. (iv) Tensor network corresponding to the expectation value $\langle\Psi|O^{[r]}|\Psi\rangle$ of an operator $O^{[r]}$, acting on site r, when $|\Psi\rangle$ is represented by an iMPS $\{\Gamma, \lambda\}$ in its canonical form. As a result of the orthonormality of the Schmidt vectors in Eq. (1), only a very small number of tensors need to be considered. (v) Tensor network for the computation of the two-point correlator $\langle\Psi|O^{[r]}O^{[s]}|\Psi\rangle$ when $|\Psi\rangle$ is represented by an iMPS $\{\Gamma, \lambda\}$ in the canonical form.

where $\eta \in \mathbb{C}$. In other words, the identity operator $I_{\alpha\alpha'} = \delta_{\alpha\alpha'}$ is a right (left) eigenvector of matrix R (respectively L) with eigenvalue η, see Fig.(1.iii). Incidentally, η is the dominant eigenvalue [15] of both R and L, and is equal to 1 if and only if $|\Psi\rangle$ is normalized [18].

Two good reasons to express an iMPS in its canonical form are the following. First, it facilitates the computation of expectation values for local operators. For instance, for $O^{[r]}$ an operator acting on site r, orthogonality of the Schmidt bases implies that $\langle\Psi|O^{[r]}|\Psi\rangle$ is simply

$$\langle\Psi|O^{[r]}|\Psi\rangle = \sum_{\alpha,i,\beta} (\lambda_\alpha)^2 \Gamma^i_{\alpha\beta} O^{[r]}_{ij} (\Gamma^j_{\alpha\beta})^* (\lambda_\alpha)^2 \, , \tag{6}$$

which can be computed in $O(d^2\chi^2)$ time. Similarly, the expression for a two-point correlator $\langle\Psi|O^{[r]}O^{[s]}|\Psi\rangle$ involves only of the order of $|s - r|$ tensors, see Figs. (1.iv)-(1.v). Second, as we will discuss in Sect. III, the canonical form simplifies the truncation of bond indices, a process that is necessary in order to prevent the rank of the iMPS from growing during a simulation.

Theorem 1 of Ref. [17] explains how to bring an MPS to its canonical form in the case of a finite chain. This is done

by orthonormalizing bond indices, starting from one boundary of the chain and progressing through the whole system, with a cost proportional to its length. In the infinite case, we use translational invariance to reduce this cost to a constant. More specifically, given an iMPS $\{\Gamma, \lambda\}$ for $|\Psi\rangle$, we can obtain a canonical form $\{\Gamma', \lambda'\}$ for the same state through three steps, illustrated in Fig. (2):

(i) Find the matrix V_R that is the dominant right eigenvector [15] of R, in the sense of Fig. (2.i), with dominant eigenvalue $\eta \in \mathbb{C}$ (here η is assumed to be unique [19]). Similarly, find the matrix V_L that is the dominant left eigenvector of L, which also has eigenvalue η [we use a large-scale, non-Hermitian eigenvalue solver [20], such as an Arnoldi method, and exploit the tensor network structure of R and L]. Decompose matrices V_R and V_L, which are Hermitian and non-negative (since they originate in the scalar product of a set of non-orthogonal vectors), as squares $V_R = XX^\dagger$ and $V_L = Y^\dagger Y$. For instance, if $V_L = WDW^\dagger$ is the eigenvalue decomposition of V_L, then $Y^\dagger = W\sqrt{D}$ and $Y = \sqrt{D}W^\dagger$ [21].

(ii) Introduce the two resolutions of the identity matrix $\mathbb{I} = (Y^T)^{-1}Y^T$ and $\mathbb{I} = XX^{-1}$ in the bond indices of the iMPS as indicated in Fig.(2.ii). Then, compute the singular value decomposition of the product $Y^T\lambda X$, namely $Y^T\lambda X = U\lambda'V$, where U, V are unitary and the diagonal matrix λ' contains the Schmidt coefficients of $|\Psi\rangle$.

(iii) Arrange the remaining tensors V, X^{-1}, Γ, $(Y^T)^{-1}$ and U into a new tensor Γ' as in Fig. (2.iii).
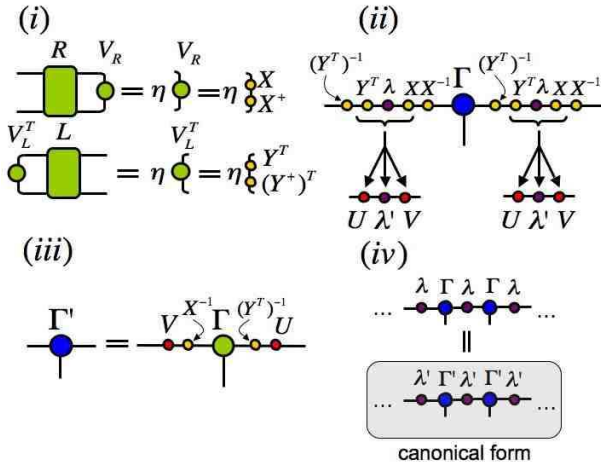

Figure 2: (color on-line) (i-iii) The three steps involved in the computation of the canonical form $\{\Gamma', \lambda'\}$ for an iMPS $\{\Gamma, \lambda\}$ for $|\Psi\rangle$, as explained in the text. In particular, λ' is obtained in step (ii) as the singular values of $Y^T\lambda X$, whereas Γ' is defined in step (iii). (iv) Both $\{\Gamma, \lambda\}$ and $\{\Gamma', \lambda'\}$ represent the same state $|\Psi\rangle$, as can be verified by direct substitution.

All the previous manipulations can be implemented with computational cost scaling as $O(d\chi^3)$. A proof that the resulting iMPS $\{\Gamma', \lambda'\}$ is indeed in the canonical form is given in Fig. (3).

We can now analyze the case where $|\Psi\rangle$ is invariant under translations by $n > 1$ sites. For $n = 2$, the state $|\Psi\rangle$ is rep-
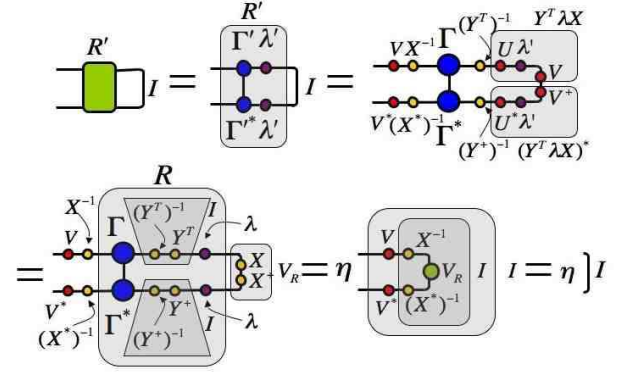

Figure 3: (color online) Proof that $\{\Gamma'\lambda'\}$, the iMPS obtained from $\{\Gamma, \lambda\}$ by re-orthonormalizing its bond indices following Fig. (2), is indeed in the canonical form. The diagram only shows the condition of Eq. (4) for R'. The proof of Eq. (5) for L' is analogous.

resented by an iMPS that consists of four alternating tensors $\{\Gamma^A, \lambda^A, \Gamma^B, \lambda^B\}$, where A and B denote odd and even sites in the chain [12]. The canonical form $\{\Gamma^{A'}, \lambda^{A'}, \Gamma^{B'}, \lambda^{B'}\}$, defined as before to correspond to the Schmidt decomposition at each bond, can be obtained as follows. First we coarse-grain the chain by regarding each pair of sites AB as a single site and represent $|\Psi\rangle$ with an iMPS (Γ, λ) as in the $n = 1$ case. Then we transform the coarse-grained iMPS (Γ, λ) into its canonical form (Γ', λ'). Finally, we split Γ' into three tensors $\Gamma^{A'}$, $\lambda^{A'}$ and $\Gamma^{B'}$ by means of a singular value decomposition, see Fig (4). These steps can be implemented with a computational cost that scales as $O(d^3\chi^3)$. The case of a generic n is addressed similarly, and the computational cost scales as $O(nd^3\chi^3)$.
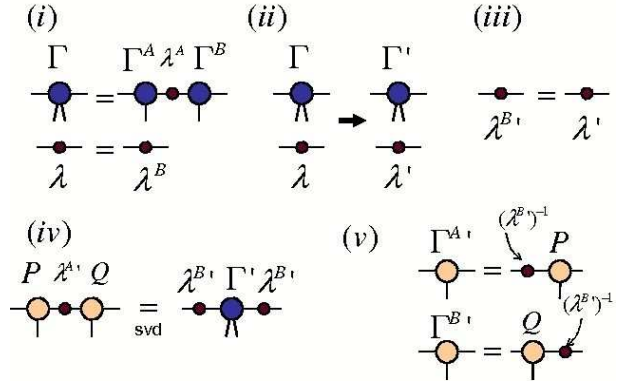

Figure 4: (color online) Construction of the canonical form $\{\Gamma^{A'}, \lambda^{A'}, \Gamma^{B'}, \lambda^{B'}\}$ from an iMPS $\{\Gamma^A, \lambda^A, \Gamma^B, \lambda^B\}$ that is invariant under translations by $n = 2$ sites: (i) Coarse-grained iMPS $\{\Gamma, \lambda\}$. (ii) Canonical form $\{\Gamma', \lambda'\}$ for the coarse-grained iMPS $\{\Gamma, \lambda\}$. (iii) $\lambda^{B'}$ corresponds to λ'. (iv) $\lambda^{A'}$ is obtained through a singular value decomposition, from where also (v) $\Gamma^{A'}$ and $\Gamma^{B'}$ are obtained after minor manipulations.

III. SIMULATION OF NON-UNITARY EVOLUTION

In this section we discuss how to update the iMPS for state $|\Psi\rangle$ after a gate G acts on the entire lattice. That is, we aim to build an iMPS for the resulting state $|\Psi'\rangle = G|\Psi\rangle$. We assume that G is expressed as a one-dimensional tensor network (of some sort) that is invariant under translations by n sites, see Fig. (5) for several examples. As a remark, let us mention that non-unitary gates such as the ones in Fig. (5) appear in the so-called iPEPS algorithm to simulate 2D quantum lattice systems [16].
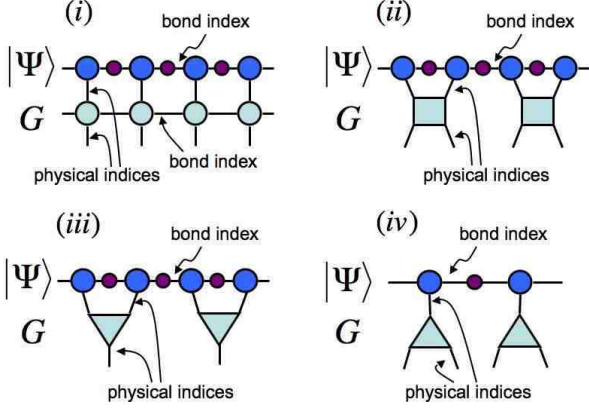

Figure 5: (color online) Four types of gates G acting on an iMPS: (i) infinite matrix product operator (iMPO); (ii) product of two-site operators; (iii) product of two-to-one-site operators; (iv) Tensor product of one-to-two-sites operators. Detailed explanations on how to update the iMPS for cases (ii)-(iv) can be found in the appendix. Notice that gates (iii) and (iv) change the number of sites in the lattice. These gates appear e.g. when coarse-graining (or fine-graining) the chain.

We focus again on the case $n = 1$ and, for concreteness, we assume G is specified by an infinite *matrix product operator* (iMPO) as in Fig. (5.i). This iMPO is represented by a tensor a of complex components $a^{ij}_{\mu\nu}$, where i and j are physical indices and μ and ν ($\mu, \nu = 1, \cdots, \kappa$) are bond indices. The update occurs in three steps, illustrated in Fig. (6):

(I) *Contraction:* the tensors $\{\Gamma, \lambda\}$ for $|\Psi\rangle$ are contracted with the tensors that specify the gate G, producing an iMPS $\{\tilde{\Gamma}, \tilde{\lambda}\}$ for $|\Psi'\rangle$,

$$\tilde{\Gamma}^{j}_{\tilde{\alpha}\tilde{\beta}} \equiv \sum_{i=1}^{d} \Gamma^{i}_{\alpha\beta} a^{ij}_{\mu\nu}, \qquad \tilde{\lambda}_{\tilde{\beta}} \equiv \lambda_{\beta} \quad (\forall\nu). \qquad (7)$$

Here indices $\tilde{\alpha}$ and $\tilde{\beta}$ ($\tilde{\alpha}, \tilde{\beta} = 1, \cdots, \tilde{\chi}$) are defined as $\tilde{\alpha} \equiv (\alpha, \mu)$ and $\tilde{\beta} \equiv (\beta, \nu)$. Notice that the rank $\tilde{\chi}$ of the new iMPS, $\tilde{\chi} \equiv \kappa\chi$, is larger than the rank χ of the initial iMPS. The computational cost of this step is $O(d^2\kappa^2\chi^2)$.

(II) *Orthogonalization:* the iMPS $\{\tilde{\Gamma}, \tilde{\lambda}\}$ for $|\Psi'\rangle$ is brought into its canonical form $\{\tilde{\Gamma}', \tilde{\lambda}'\}$ with a cost that scales as $O(d\kappa^3\chi^3)$.

(III) *Truncation:* a final iMPS $\{\Gamma', \lambda'\}$ is obtained from the canonical form $\{\tilde{\Gamma}', \tilde{\lambda}'\}$ by truncating all bond indices. In

particular, on each bond we preserve the first χ values of the index, corresponding to the χ largest Schmidt coefficients.
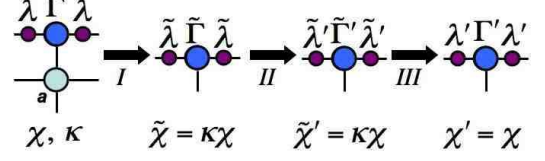

Figure 6: (color online) Sequence of transformations (I)-(III) that produce a truncated iMPS $\{\Gamma'\lambda'\}$ for $|\Psi'\rangle = G|\Psi\rangle$ from an iMPS $\{\Gamma, \lambda\}$ for $|\Psi\rangle$, as explained in the text. Notice in particular that, as a result of the truncation step, the initial and final iMPSs have the same rank χ.

The net result is an *approximate* iMPS $\{\Gamma', \lambda'\}$ for $|\Psi'\rangle$, obtained with a total computational cost $O(d^2\kappa^2\chi^2 + d\kappa^3\chi^3)$ [22]. The truncation step is necessary in order to keep the rank χ (and therefore the computational cost) constant during a simulation, where typically not just one gate G but rather of a whole series $\{G_1, G_2, \cdots\}$ is sequentially applied to the chain. The truncation of the bond indices introduces an error that is hard to evaluate in an infinite system. Here we apply, to all bond indices simultaneously, the truncation scheme that is known to be optimal when applied only to one bond index [23].

For $n > 1$, as is the case e.g. in Fig. (5.ii), we can again coarse-grain the system and proceed as in the $n = 1$ case, which will result in a iMPS $\{\Gamma', \lambda'\}$. Then Γ' is broken into several other tensors $\{\Gamma^{A'}, \lambda^{A'}, \Gamma^{B'}, \cdots\}$, process that may require additional truncations. See the appendix for a detailed analysis of some particular cases.

We are therefore able to address non-unitary evolution on an infinite chain. When the gate G breaks into a row of two-site gates as in Fig. (5.ii), and each two-site gate is unitary, then the canonical form of the iMPS is preserved (up to truncation errors) without need of the orthogonalization step, recovering the original formulation of the iTEBD algorithm [12]. Notice that in Ref. [12] the algorithm is also used to compute the ground state of the system. This is done by simulating (non-unitary) imaginary time evolution

$$|\Psi_\tau\rangle = \frac{\exp(-H\tau)|\Psi_0\rangle}{||\exp(-H\tau)|\Psi_0\rangle||}, \qquad (8)$$

where H is the Hamiltonian of the infinite chain and $|\Psi_0\rangle$ some initial state, and by exploiting the fact that under proper circumstances the ground state $|\Psi\rangle$ of H is the fixed point of such evolution,

$$|\Psi\rangle = \lim_{\tau\to\infty} \frac{\exp(-H\tau)|\Psi_0\rangle}{||\exp(-H\tau)|\Psi_0\rangle||}. \qquad (9)$$

We emphasize that such calculation succeeds thanks to a fortunate combination of favorable, unlikely circumstances. When the simulation is performed using small time steps, two-site gates that are close to the identity operator are used. These gates destroy the canonical form of an initial iMPS, but they

leave its bond indices in a (non-orthonormal) basis that still seems to lead to reasonably small errors during their truncation. One would expect the bond bases to become less and less adequate for truncation over time, as the accumulated τ increases, since the overall evolution $\exp(-H\tau)$ departs more and more from the identity. But it turns out that the singular value decomposition used in order to update the iMPS at each time step has the effect of reorganizing the indices favorably, partially compensating the non-unitary effects [24]. Finally, all the excessive truncation errors introduced during the simulation are washed away at its final stages, where increasingly small time steps are used. These have the intended effect of reducing Suzuki-Trotter errors [3], but they also imply that the gates become almost unitary (that is, very close to the identity). One can see that, as a result, by the end of the simulation the iMPS approximation for the ground state is not only accurate, but it is also very close to the canonical form.

Thanks to actively transforming the iMPS into its canonical form, the present extension is not restricted to unitary evolution and can be applied to a wider range of 1D problems. In particular, it can be used to simulate master equation evolution and to compute thermal states using the mixed state formalism of Refs. [13]. Importantly, it can also be used to manipulate the state of an infinite 2D lattice, both for classical (see example below) and quantum systems [16]. This is achieved after the 2D problem is recast into that of finding the dominant eigenvalue θ and dominant eigenvector [15] $|\Psi\rangle$ of a 1D transfer matrix T that decomposes into a finite sequence $\{G_1, G_2, \cdots, G_m\}$ of gates. The dominant eigenvector satisfies

$$|\Psi\rangle = \lim_{p \to \infty} \frac{T^p |\Psi_0\rangle}{||T^p |\Psi_0\rangle||}, \quad (10)$$

and is obtained by simulating the repeated application of T on an initial state $|\Psi_0\rangle$, until converge is attained. The dominant eigenvalue can be obtained from the dominant eigenvector $|\Psi\rangle$ and any other vector $|\Phi\rangle$, $\langle\Phi|\Psi\rangle \neq 0$, since

$$\theta = \frac{\langle\Phi|T|\Psi\rangle}{\langle\Phi|\Psi\rangle}. \quad (11)$$

In the next section we provide an explicit example of calculation of dominant eigenvalue and eigenvector of a one-dimensional transfer matrix.

IV. EXAMPLE: 2D CLASSICAL SYSTEMS

In this section we explain how the above algorithm can be used to compute the partition function, local observables and two-point correlators of a classical spin system. We consider an infinite 2D lattice where each site, labeled by a vector $\vec{r}$, contains a d-dimensional spin $s^{[\vec{r}]}$ that interacts with nearest neighbor spins according to a Hamiltonian K,

$$K(\{s\}) = \sum_{\langle \vec{r}, \vec{r}' \rangle} K_2(s^{[\vec{r}]}, s^{[\vec{r}']}). \quad (12)$$

The system's partition function reads

$$Z(\beta) = \sum_{\{s\}} e^{-\beta K(\{s\})} = \sum_{\{s\}} \prod_{\langle \vec{r}, \vec{r}' \rangle} e^{-\beta K_2(s^{[\vec{r}]}, s^{[\vec{r}']})}, \quad (13)$$

where β is the inverse temperature. For concreteness, we consider a square lattice with an isotropic interaction K_2. Let $\sqrt{Q}$ denote the squared root of the Hermitian matrix $Q_{ss'} \equiv \exp(-\beta K_2(s, s'))$ [25]. We can express the partition function as the contraction of an infinite 2D tensor network specified by a single tensor a,

$$a_{ijkl} \equiv \sum_s (\sqrt{Q})_{is} (\sqrt{Q})_{js} (\sqrt{Q})_{qs} (\sqrt{Q})_{ls} \quad (14)$$

that is repeated on all sites [26], see Fig. (7). The above tensor can be computed in $O(d^5)$ time.
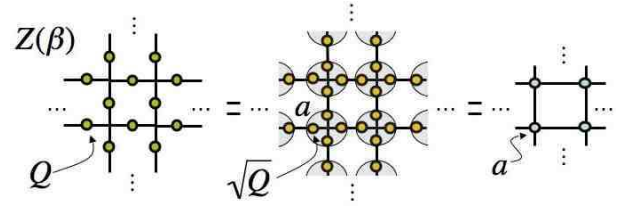

Figure 7: (color online) The partition function $Z(\beta)$ of a 2D classical system can be written as the contraction of a 2D tensor network. In the case of an infinite square lattice with isotropic and homogeneous interactions, this tensor network consists of infinitely many copies of the tensor a in Eq. (14).

We now introduce an infinite 1D transfer matrix T consisting of one row of tensors a, see Fig. (8). Then we have

$$Z(\beta) = \lim_{p \to \infty} \text{tr}(T^p) = \lim_{p \to \infty} \theta^p, \quad (15)$$

where θ is the dominant eigenvalue of T. Let $|\Psi_U\rangle$ and $|\Psi_D\rangle$ be the corresponding (up and down) eigenvectors,

$$T|\Psi_U\rangle = \theta|\Psi_U\rangle, \qquad \langle\Psi_D|T = \theta\langle\Psi_D|, \quad (16)$$

that we normalize to $\langle\Psi_D|\Psi_U\rangle = 1$. Then

$$\theta = \langle\Psi_D|T|\Psi_U\rangle = \text{tr}(W^q) = \lim_{q \to \infty} \omega^q, \quad (17)$$

where ω is the dominant eigenvalue of matrix W defined in Fig. (8), and we finally have

$$Z(\beta) = \lim_{p,q \to \infty} \omega^{p\,q}. \quad (18)$$

Therefore, in order to evaluate the partition function $Z(\beta)$, we will first construct an iMPS $\{\Gamma^U, \lambda^U\}$ for $|\Psi_U\rangle$ and an iMPS $\{\Gamma^D, \lambda^D\}$ for $|\Psi_D\rangle$ by iteratively applying the transfer matrix T on an initial state $|\Psi_0\rangle$ (c.f. Eq. (10)). Specifically, we use the iTEBD algorithm as discussed in the previous section to simulate the state

$$|\Psi_p\rangle \equiv \frac{T^p |\Psi_0\rangle}{||T^p |\Psi_0\rangle||}, \qquad p = 1, 2, \cdots \quad (19)$$

for increasing values of p, until the resulting iMPS has converged within some agreed precision [27]. The computation is approximate, in that an iMPS with finite rank χ will be used to represent the dominant eigenvectors, which in general may only be represented exactly with an infinite rank χ. Notice that if K_2 is isotropic, the transfer matrix can be made Hermitian, in which case $\langle \Psi_D | = |\Psi_U^\dagger \rangle$ [otherwise $|\Psi_D\rangle$ also needs to be computed]. From the converged iMPSs $\{\Gamma^U, \lambda^U\}$ and $\{\Gamma^D, \lambda^D\}$ we can construct matrix W. The dominant eigenvalue ω of W can then be computed using a large-scale eigenvalue solver and exploiting its tensor network structure in $O(d^2\chi^3 + d^4\chi^2)$ time.
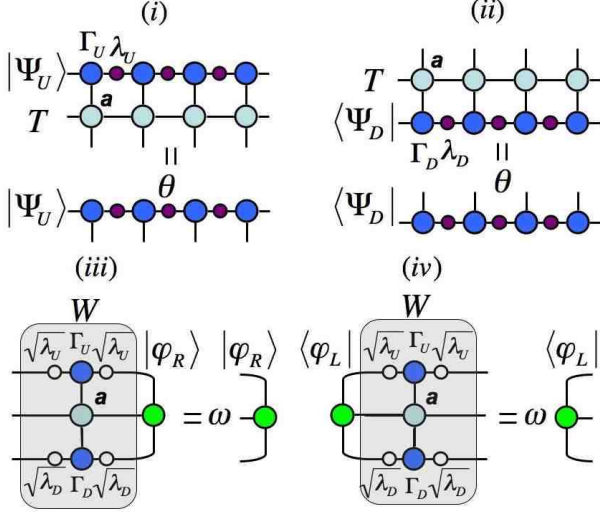

Figure 8: (color online) Computation of the partition function through Eq. (18). (i) $|\Psi_U\rangle$ is the up dominant eigenvector of the transfer matrix T, with dominant eigenvalue θ. (ii) Similarly, $|\Psi_D\rangle$ is a down dominant eigenvector of T, with dominant eigenvalue θ. (iii) $|\varphi_R\rangle$ is the right dominant eigenvector of matrix W, with dominant eigenvalue ω. (iv) $|\varphi_L\rangle$ is the left dominant eigenvector of matrix W, with dominant eigenvalue ω.

On the other hand, for any function $f(s)$ of one spin, the expectation value

$$\langle f(s^{[\vec{r}]})\rangle = \frac{1}{Z(\beta)} \sum_{\{s\}} f(s^{[\vec{r}]}) e^{-\beta K(\{s\})} , \qquad (20)$$

is, up to the factor $1/Z(\beta)$, also given by the contraction of an infinite 2D tensor network, obtained from that for $Z(\beta)$ by replacing tensor a on site $\vec{r}$ with tensor b,

$$b_{ijkl} \equiv \sum_s f(s)(\sqrt{Q})_{is}(\sqrt{Q})_{js}(\sqrt{Q})_{qs}(\sqrt{Q})_{ls} , \qquad (21)$$

again computable in $O(d^5)$ time. As illustrated in Fig. (9), $\langle f(s^{[\vec{r}]})\rangle$ is eventually written as the ratio of two small tensor networks. These tensor networks are expressed entirely in terms of: tensors a and b; tensors $\{\Gamma^U, \lambda^U\}$ and $\{\Gamma^D \lambda^D\}$ defining the dominant eigenvectors $|\Psi_U\rangle$ and $|\Psi_D\rangle$ of the one-dimensional transfer matrix T; and the dominant vectors $|\varphi_R\rangle$
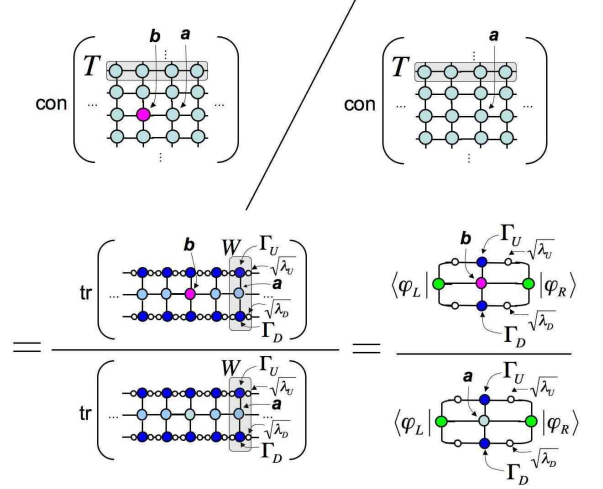

Figure 9: (color online) The expectation value $\langle f(s^{[\vec{r}]})\rangle$ from Eq. (20) is the ratio of the contraction of two infinite 2D tensor networks. By introducing the dominant eigenvectors of the one-dimensional transfer matrix T, we can rewrite $\langle f(s^{[\vec{r}]})\rangle$ as the ratio of the trace of two infinite 1D tensor networks. Finally, by introducing the dominant eigenvectors of matrix W, we obtain a ration of two simple tensor networks.

and $|\varphi_L\rangle$ of the matrix W. We have already indicated how to proceed in the computation of these quantities.

Similarly, we can build a tensor network for the expectation value of the correlator

$$\langle f(s^{[\vec{r}]}) g(s^{[\vec{r}']})\rangle = \frac{1}{Z(\beta)} \sum_{\{s\}} f(s^{[\vec{r}]}) g(s^{[\vec{r}']}) e^{-\beta K(\{s\})} , \qquad (22)$$

by replacing the tensor a in sites $\vec{r}$ and $\vec{r}'$ with appropriate tensors b and b' and proceeding in a similar way as the previous case, see Fig. (10). Notice that we assume that sites $\vec{r}$ and $\vec{r}'$ lie on the same row of the lattice.
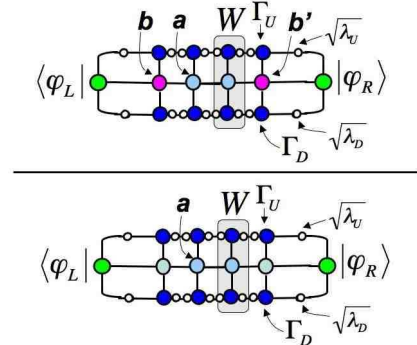

Figure 10: (color online) Following steps analogous as those of Fig. (9) for the expectation value $\langle f(s^{[\vec{r}]})\rangle$, the two-point correlator $\langle f(s^{[\vec{r}]}) g(s^{[\vec{r}']})\rangle$ can also be reduced to the ratio of two simple tensor networks.

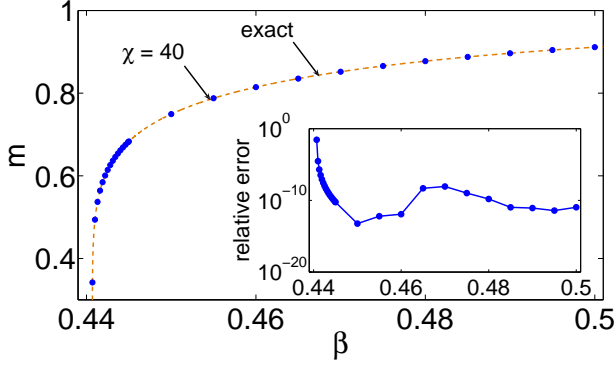Fig. (11) shows the magnetization per site $m \equiv \langle s^{[\vec{r}]}\rangle$ for

Figure 11: (color online) Magnetization per lattice site for the infinite-size 2D classical Ising model at different temperatures β. The exact solution $m = (1-((\sinh{(2\beta)})^{-4}))^{1/8}$ has been included. The numerical results have been obtained by approximating the dominant eigenvectors of the one-dimensional transfer matrix T with an iMPS of rank $\chi = 40$. The inset shows the relative error.
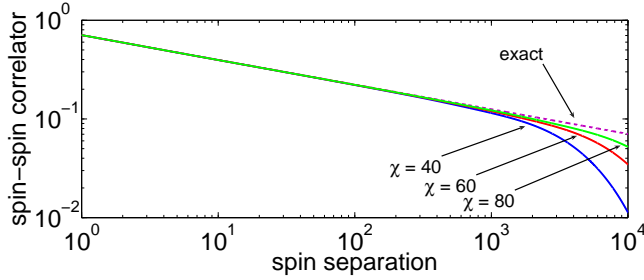

Figure 12: (color online) Two-point correlators for the infinite-size 2D classical Ising model at critical temperature, $\beta_c = \frac{1}{2}\log(1+\sqrt{2})$, along a row of the square lattice. The exact solution, which scales as $\langle s^{[\vec{r}]}s^{[\vec{r}']}\rangle_{\beta_c} \approx c|\vec{r} - \vec{r}'|^{-1/4}$, has been included. The numerical results have been obtained by approximating the dominant eigenvectors of the one-dimensional transfer matrix T with an iMPS of rank $\chi = 40$, $\chi = 60$ and $\chi = 80$.

the 2D Ising model, defined by

$$K_2(s, s') = -s\,s' \qquad s, s' = \pm 1 , \qquad (23)$$

at different values of β. We have used an iMPS of rank $\chi = 40$ to represent the dominant eigenvectors $|\Psi_U\rangle$ and $|\Psi_D\rangle$, and proceeded as explained above. It is noteworthy that the numerical results reproduce the exact behaviour of m with small relative error. Furthermore, Fig. (12) shows the decay with $|\vec{r} - \vec{r}'|$ of the spin-spin correlator $\langle s^{[\vec{r}]}s^{[\vec{r}']}\rangle$ at the critical point, $\beta_c = \frac{1}{2}\log(1 + \sqrt{2})$. In this case we have used an iMPS of rank $\chi = 40, 60, 80$. Remarkably, the numerical results reproduce the correct power-law decay $\sim |\vec{r}-\vec{r}'|^{-1/4}$ for distances up to thousands of spins, with increasing accuracy as χ increases.

V. CONCLUSIONS

In this paper we have explained how to extend the iTEBD algorithm so that it can be applied to simulate any evolution that can be expressed as a sequence of one-dimensional tensor networks. The key new ingredient is a recipe to rewrite any iMPS in the canonical form, as required in order to properly truncate the bond indices.

The iTEBD algorithm can therefore be applied to simulate not only unitary evolution, but also master equation evolution and imaginary time evolution. It can also be used to find the dominant eigenvalue and dominant eigenvector of a one-dimensional transfer matrix. This last application is particularly relevant in order to analyze the partition function of 2D classical models, as we explained, and it also plays a prominent role in the iPEPS algorithm for 2D quantum systems [16] and some of the MERA algorithms [6].

Appendix A: ALGORITHMS FOR THE EVOLUTIONS IN FIG.(5)

In this appendix we explain in some detail how to implement the iTEBD algorithm for some particularly relevant choices of the gate G. In Sect. III we analyzed the case where the system is invariant under translations by $n = 1$ lattice sites. Here we consider the case $n = 2$ for the four different types of gate represented in Fig.(5). Most of the information contained in the appendix can already be derived from the results of Sects. II and III, but we write it explicitly for each gate for the sake of clarity.

1. Matrix product operator, Fig.(5.*i*)

Let us assume that the iMPS for $|\Psi\rangle$ is given by some tensors Γ^A, λ^A for odd sites and tensors Γ^B, λ^B for even sites. We consider the evolution under a MPO invariant under translations of two chain sites,

$$G = \sum_{\alpha,\beta,\gamma,\delta,\rho,\ldots} \cdots a_{\alpha\beta}^{[i]}b_{\beta\gamma}^{[i+1]}a_{\gamma\delta}^{[i+2]}b_{\delta\rho}^{[i+3]}\cdots , \qquad (A1)$$

where for each value of $\alpha, \beta = 1, 2, \ldots, \kappa$ the one-site operators $a_{\alpha\beta}^{[i]} : \mathcal{H}^{[i]} \to \mathcal{H}^{[i]}$ and $b_{\alpha\beta}^{[i+1]} : \mathcal{H}^{[i+1]} \to \mathcal{H}^{[i+1]}$ act on sites i and $i + 1$, see Fig.(5.*i*). The updating procedure of the iMPS can be expressed as follows:

(*i*) Compute tensor Θ_1 as indicated in Fig.(13.*i*), with bond dimension $\kappa\chi$.

(*ii*) Find the matrix V_R that is the right dominant [15] eigenvector of R (in the sense of Fig.(13.*ii*)) with dominant eigenvalue $\eta \in \mathbb{C}$ (assumed to be unique) [19], where R is

obtained by contracting Θ_1 with its complex conjugate Θ_1^* as shown in Fig.($13.ii$) [use large-scale, non-Hermitian eigenvalue solver, such as Arnoldi methods, and exploit the tensor network structure of R]. Then, decompose matrix V_R (which is Hermitian and non-negative) as the square $V_R = XX^\dagger$. For instance, if $V_R = WDW^\dagger$ is the eigenvalue decomposition of V_R, then $X = W\sqrt{D}$.

(iii) Compute tensor Θ_2 as indicated in Fig.($13.iii$), with bond dimension $\kappa\chi$.

(iv) Find the matrix V_L that is the left dominant [15] eigenvector of L (in the sense of Fig.($13.iv$)) with dominant eigenvalue $\tau \in \mathbb{C}$ (assumed to be unique) [19], where L is obtained by contracting Θ_2 with its complex conjugate Θ_2^* as shown in Fig.($13.iv$) [use large-scale, non-Hermitian eigenvalue solver, such as Arnoldi methods, and exploit the tensor network structure of L]. Then, decompose matrix V_L (which is Hermitian and non-negative) as the square $V_L = Y^\dagger Y$, in the same way as was done for matrix V_R.

(v) Compute tensor Θ as indicated in Fig.($13.v$), with bond dimension $\kappa\chi$.

(vi) Introduce the two resolutions of the identity matrix $\mathbb{I} = (Y^T)^{-1}Y^T$ and $\mathbb{I} = XX^{-1}$ in the bond indices of tensor Θ as indicated in Fig.($13.vi$). Then, compute the singular value decomposition $Y^T X = U\lambda^{B\prime}V$, leading to new Schmidt coefficients $\lambda^{B\prime}$. Truncate these new Schmidt coefficients by keeping only the χ largest ones, and normalize them so that the sum of their squared values is 1.

(vii) Compute tensor Σ as indicated in Fig.($13.vii$).

(iix) Group the indices of Σ according to a single index for the left-hand side and a single index for the right-hand side, and compute the singular value decomposition as indicated in Fig.($13.iix$). This leads to two isometric tensors P and Q, and new Schmidt coefficients $\lambda^{A\prime}$. Truncate these new Schmidt coefficients by keeping only the χ largest ones, and normalize them so that the sum of their squared values is 1.

(ix) Obtain new matrices $\Gamma^{A\prime}$ and $\Gamma^{B\prime}$ as indicated in Fig.($13.ix$).

The above sequence of steps has a computational cost of $O(d^2\kappa^3\chi^3)$ in time.

2. Tensor product of two-site operators, Fig.($5.ii$)

Consider an iMPS for state $|\Psi\rangle$ with bond dimension χ that is invariant under shifts of two chain sites. This iMPS is then defined by tensors Γ^A, λ^A for odd sites and tensors Γ^B, λ^B for even sites. The evolution operator that we consider is given by

$$G = \bigotimes_{i\in \text{ odd}} a^{[i,i+1]}, \qquad (A2)$$

where $a^{[i,i+1]} : \mathcal{H}^{[i]} \otimes \mathcal{H}^{[i+1]} \to \mathcal{H}^{[i]} \otimes \mathcal{H}^{[i+1]}$ is a two-body operator acting on the two contiguous sites i and $i+1$ of the iMPS, see Fig.($5.ii$). The algorithm to update the iMPS is as follows:

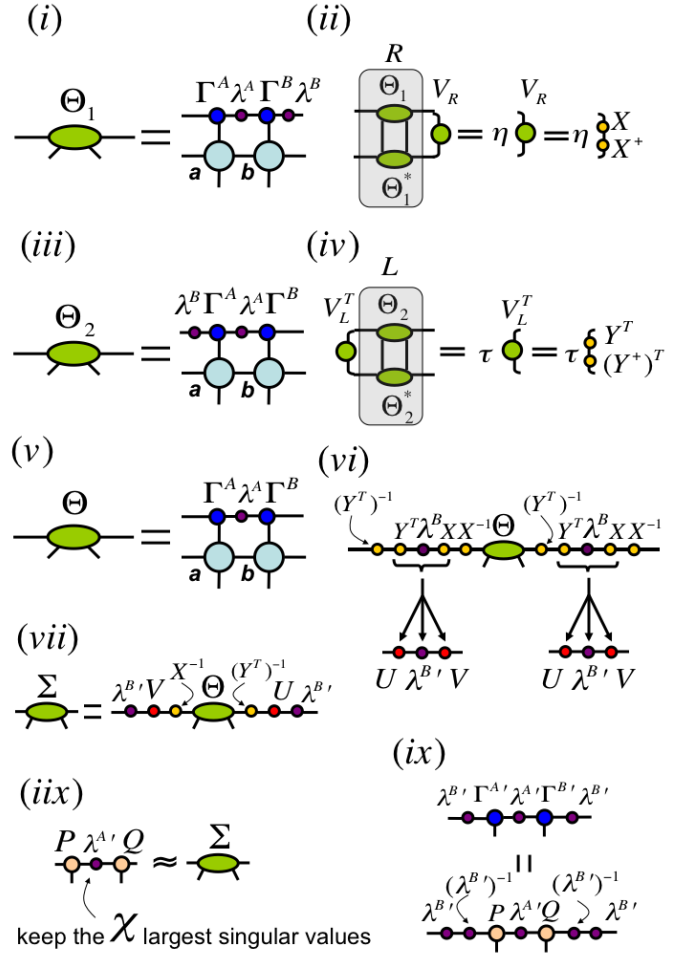(i) Compute tensor Θ_1 as indicated in Fig.($14.i$), with bond dimension χ.

Figure 13: (color online) Steps in the updating of the iMPS after the action of a MPO.

(ii) Find the matrix V_R that is the right dominant [15] eigenvector of R (in the sense of Fig.($14.ii$)) with dominant eigenvalue $\eta \in \mathbb{C}$ (assumed to be unique) [19], where R is obtained by contracting Θ_1 with its complex conjugate Θ_1^* as shown in Fig.($14.ii$) [use large-scale, non-Hermitian eigenvalue solver, such as Arnoldi methods, and exploit the tensor network structure of R]. Then, decompose matrix V_R (which is Hermitian and non-negative) as the square $V_R = XX^\dagger$. For instance, if $V_R = WDW^\dagger$ is the eigenvalue decomposition of V_R, then $X = W\sqrt{D}$.

(iii) Compute tensor Θ_2 as indicated in Fig.($14.iii$), with bond dimension χ.

(iv) Find the matrix V_L that is the left dominant [15] eigenvector of L (in the sense of Fig.($14.iv$)) with dominant eigenvalue $\tau \in \mathbb{C}$ (assumed to be unique) [19], where L is obtained by contracting Θ_2 with its complex conjugate Θ_2^* as shown in Fig.($14.iv$) [use large-scale, non-Hermitian eigenvalue solver, such as Arnoldi methods, and exploit the tensor network structure of L]. Then, decompose matrix V_L (which is Hermitian and non-negative) as the square $V_L = Y^\dagger Y$, in the same way as was done for matrix V_R.

(v) Compute tensor Θ as indicated in Fig.($14.v$), with bond dimension χ.

(vi) Introduce the two resolutions of the identity matrix $\mathbb{I} = (Y^T)^{-1}Y^T$ and $\mathbb{I} = XX^{-1}$ in the bond indices of tensor Θ as indicated in Fig.($14.vi$). Then, compute the singular value decomposition $Y^TX = U\lambda^{B\prime}V$, leading to new Schmidt coefficients $\lambda^{B\prime}$. Truncate these new Schmidt coefficients by keeping only the χ largest ones, and normalize them so that the sum of their squared values is 1.

(vii) Compute tensor Σ as indicated in Fig.($14.vii$).

(iix) Group the indices of Σ according to a single index for the left-hand side and a single index for the right-hand side, and compute the singular value decomposition as indicated in Fig.($14.iix$). This leads to two isometric tensors P and Q, and new Schmidt coefficients $\lambda^{A\prime}$. Truncate these new Schmidt coefficients by keeping only the χ largest ones, and normalize them so that the sum of their squared values is 1.

(ix) Obtain new matrices $\Gamma^{A\prime}$ and $\Gamma^{B\prime}$ as indicated in Fig.($14.ix$).

The computational cost of the above sequence of steps is $O(d^4\chi^3)$. Also, and as expected, if $a^{[i,i+1]}$ is a unitary operator then this procedure corresponds exactly to the updating rules of the standard iTEBD algorithm.

3. Tensor product of two-to-one-site operators, Fig.($5.iii$)

Our concern now is the evolution of an iMPS under an operator G that is the tensor product of two-to-one-site operators

$$G = \bigotimes_{i \in \text{ odd}} a^{[i,i+1]} , \qquad (A3)$$

where $a^{[i,i+1]} : \mathcal{H}^{[i]} \otimes \mathcal{H}^{[i+1]} \to \mathcal{H}^{[(i+1)/2]}$ is a two-to-one-site operator acting on two contiguous sites i and $i+1$ of the iMPS, and which maps the two sites to a new site $(i+1)/2$, see Fig.($5.iii$). Again we assume that the iMPS is defined by Γ^A, λ^A for odd sites and Γ^B, λ^B for even sites. The algorithm to update the iMPS is as follows:

(i) Compute tensor Θ_1 as indicated in Fig.($15.i$), with bond dimension χ.

(ii) Find the matrix V_R that is the right dominant [15] eigenvector of R (in the sense of Fig.($15.ii$)) with dominant eigenvalue $\eta \in \mathbb{C}$ (assumed to be unique) [19], where R is obtained by contracting Θ_1 with its complex conjugate Θ_1^* as shown in Fig.($15.ii$) [use large-scale, non-Hermitian eigenvalue solver, such as Arnoldi methods, and exploit the tensor network structure of R]. Then, decompose matrix V_R (which is Hermitian and non-negative) as the square $V_R = XX^\dagger$. For instance, if $V_R = WDW^\dagger$ is the eigenvalue decomposition of V_R, then $X = W\sqrt{D}$.

(iii) Compute tensor Θ_2 as indicated in Fig.($15.iii$), with bond dimension χ.

(iv) Find the matrix V_L that is the left dominant [15] eigenvector of L (in the sense of Fig.($15.iv$)) with dominant eigenvalue $\tau \in \mathbb{C}$ (assumed to be unique) [19], where L is obtained by contracting Θ_2 with its complex conjugate Θ_2^* as shown in
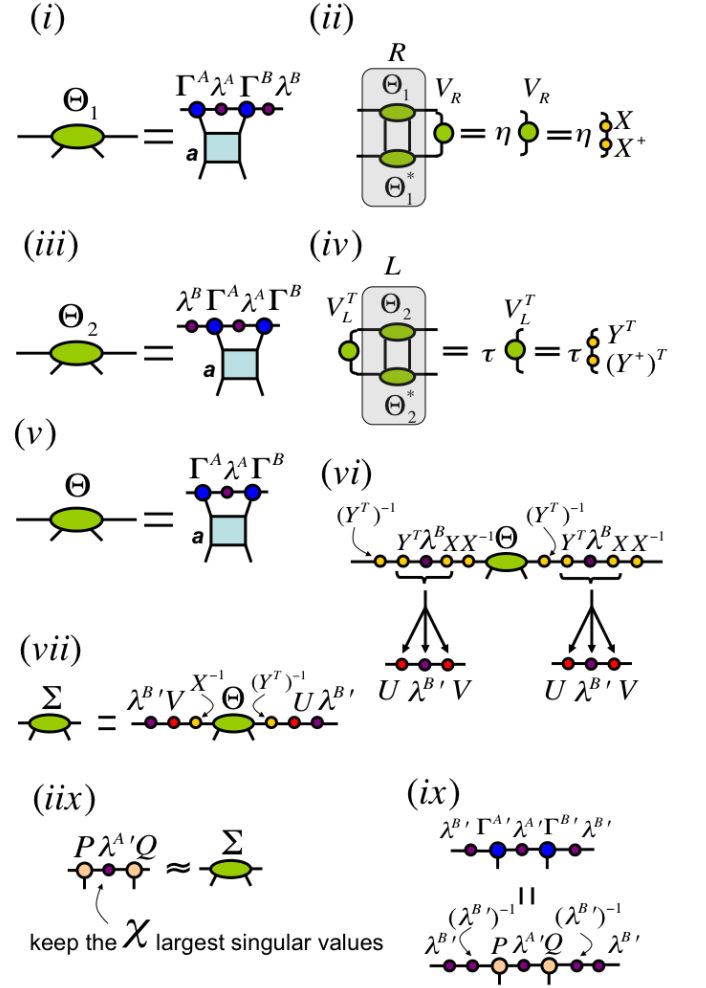

Figure 14: (color online) Steps in the updating of the iMPS after the action of the tensor product of two-site operators.

Fig.($15.iv$) [use large-scale, non-Hermitian eigenvalue solver, such as Arnoldi methods, and exploit the tensor network structure of L]. Then, decompose matrix V_L (which is Hermitian and non-negative) as the square $V_L = Y^\dagger Y$, in the same way as was done for matrix V_R.

(v) Compute tensor Θ as indicated in Fig.($15.v$), with bond dimension χ.

(vi) Introduce the two resolutions of the identity matrix $\mathbb{I} = (Y^T)^{-1}Y^T$ and $\mathbb{I} = XX^{-1}$ in the bond indices of tensor Θ as indicated in Fig.($15.vi$). Then, compute the singular value decomposition $Y^TX = U\lambda'V$, leading to new Schmidt coefficients λ'. Truncate these new Schmidt coefficients by keeping only the χ largest ones, and normalize them so that the sum of their squared values is 1.

(ix) Obtain a new matrix Γ' as indicated in Fig.($15.vii$).

The above procedure has a computational cost of $O(d^3\chi^3)$. In the end, the action of the two-to-one-site gates $a^{[i]}$ on the iMPS can be computed exactly without further truncation of the bond indices, and is such that the obtained iMPS for the evolved state $|\Psi'\rangle$ is invariant under translations of one chain
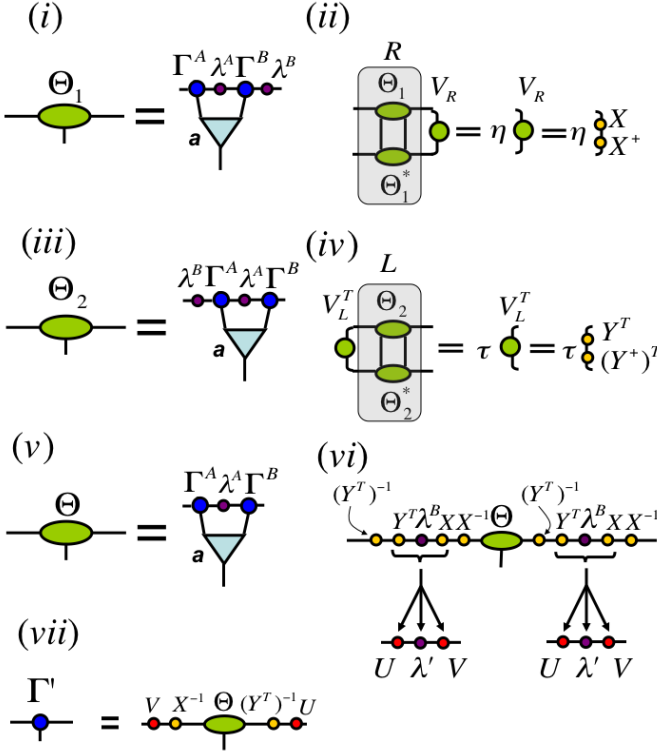
Figure 15: (color online) Steps in the updating of the iMPS after the action of the tensor product of two-to-one-site gates.

site instead of two.

4. Tensor product of one-to-two-sites operators, Fig.(5.iv)

Contrary to the previous cases, we consider now the situation in which the iMPS for state $|\Psi\rangle$ is defined by one tensor Γ and one Schmidt vector λ, so that it is invariant under shifts of one chain site. At this point we wish to update the iMPS after the action of a tensor product of one-to-two-sites operators G

$$G = \bigotimes_i a^{[i]} , \qquad (A4)$$

where $a^{[i]} : \mathcal{H}^{[i]} \rightarrow \mathcal{H}^{[2i-1]} \otimes \mathcal{H}^{[2i]}$ is a one-to-two-sites operator acting on one site i of the iMPS, and which maps the site to two new sites $2i-1$ and $2i$, see Fig.(5.iv). The steps to follow to update the iMPS are:

(i) Compute tensor Θ_1 as indicated in Fig.(16.i), with bond dimension χ.

(ii) Find the matrix V_R that is the right dominant [15] eigenvector of R (in the sense of Fig.(16.ii)) with dominant eigenvalue $\eta \in \mathbb{C}$ (assumed to be unique) [19], where R is obtained by contracting Θ_1 with its complex conjugate Θ_1^* as shown in Fig.(16.ii) [use large-scale, non-Hermitian eigenvalue solver, such as Arnoldi methods, and exploit the tensor network structure of R]. Then, decompose matrix V_R (which is Hermitian and non-negative) as the square $V_R = XX^\dagger$. For
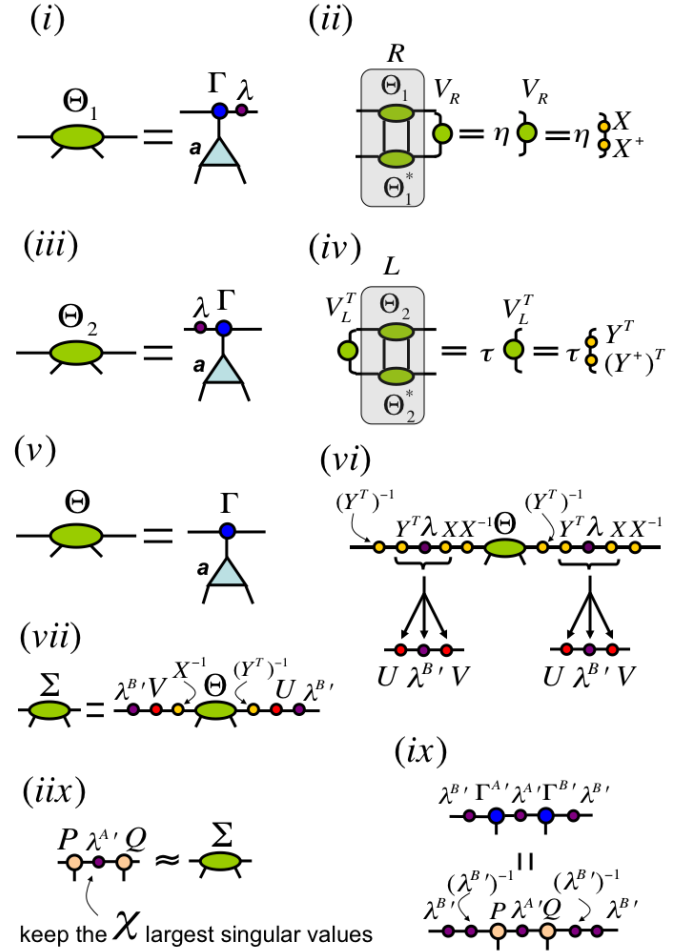

Figure 16: (color online) Steps in the updating of the iMPS after the action of the tensor product of one-to-two-sites gates.

instance, if $V_R = WDW^\dagger$ is the eigenvalue decomposition of V_R, then $X = W\sqrt{D}$.

(iii) Compute tensor Θ_2 as indicated in Fig.(16.iii), with bond dimension χ.

(iv) Find the matrix V_L that is the left dominant [15] eigenvector of L (in the sense of Fig.(16.iv)) with dominant eigenvalue $\tau \in \mathbb{C}$ (assumed to be unique) [19], where L is obtained by contracting Θ_2 with its complex conjugate Θ_2^* as shown in Fig.(16.iv) [use large-scale, non-Hermitian eigenvalue solver, such as Arnoldi methods, and exploit the tensor network structure of L]. Then, decompose matrix V_L (which is Hermitian and non-negative) as the square $V_L = Y^\dagger Y$, in the same way as was done for matrix V_R.

(v) Compute tensor Θ as indicated in Fig.(16.v), with bond dimension χ.

(vi) Introduce the two resolutions of the identity matrix $\mathbb{I} = (Y^T)^{-1}Y^T$ and $\mathbb{I} = XX^{-1}$ in the bond indices of tensor Θ as indicated in Fig.(16.vi). Then, compute the singular value decomposition $Y^T X = U\lambda^{B\prime}V$, leading to new Schmidt coefficients $\lambda^{B\prime}$. Truncate these new Schmidt coefficients by keeping only the χ largest ones, and normalize them

so that the sum of their squared values is 1.

(vii) Compute tensor Σ as indicated in Fig.($16.vii$).

(iix) Group the indices of Σ according to a single index for the left-hand side and a single index for the right-hand side, and compute the singular value decomposition as indicated in Fig.($16.iix$). This leads to isometric two tensors P and Q, and new Schmidt coefficients $\lambda^{A\prime}$. Truncate these new Schmidt coefficients by keeping only the χ largest ones, and normalize them so that the sum of their squared values is 1.

(ix) Obtain new matrices $\Gamma^{A\prime}$ and $\Gamma^{B\prime}$ as indicated in Fig.($16.ix$).

The computational cost of the above steps is $O(d^3\chi^3)$. Similarly to the case of the previous section, the translational invariance of the original iMPS for $|\Psi\rangle$ has been modified, in a way that the obtained iMPS representation for the evolved state $|\Psi'\rangle$ has periodicity under shifts of two chain sites instead of one.

[1] S. Ostlund and S. Rommer, Phys. Rev. Lett. 75, 3537 (1995). M. Fannes, B. Nachtergaele, R. Werner, Commun. Math. Phys. 144, 443 (1992). D. Perez-Garcia, F. Verstraete, M.M. Wolf, J.I. Cirac, Quantum Inf. Comput. 7, 401 (2007)

[2] S. R. White, Phys. Rev. Lett. 69, 2863 (1992). S.R.White, Phys. Rev. B 48, 10345 (1992).

[3] G. Vidal, Phys. Rev. Lett. 91, 147902 (2003). G. Vidal, Phys. Rev. Lett. 93, 040502 (2004). S. R. White, A. E. Feiguin, Phys. Rev. Lett. 93, 076401 (2004). A. J. Daley, C. Kollath, U. Schollwoeck, G. Vidal, J. Stat. Mech.: Theor. Exp. (2004) P04005.

[4] T. Nishino, K. Okunishi, Y. Hieida, N. Maeshima, Y. Akutsu, Nucl. Phys. B 575 (2000) 504-512. T. Nishino, Y. Hieida, K. Okunishi, N. Maeshima, Y. Akutsu, A. Gendiar, Prog. Theor. Phys. 105 (2001) No.3, 409-417. A. Gendiar, N. Maeshima, T. Nishino, Prog. Theor. Phys. 110 (2003) No.4, 691-699. N. Maeshima, Y. Hieida, Y. Akutsu, T. Nishino, K. Okunishi, Phys. Rev. E64 (2001) 016705 [1-6]. Y. Nishio, N. Maeshima, A. Gendiar, T. Nishino, cond-mat/0401115. A. Gendiar, T. Nishino, R. Derian, Acta Phys. Slov. 55 (2005) 141.

[5] F. Verstraete, J. I. Cirac, cond-mat/0407066. V. Murg, F. Verstraete, J. I. Cirac, Phys. Rev. A 75, 033605 (2007)

[6] G. Vidal, Phys. Rev. Lett. 99, 220405 (2007). G. Vidal, arXiv:0707.1454

[7] T. Nishino, J. Phys. Soc. Jpn. 64 (1995) 3598-3601. T. Nishino, K. Okunishi, "Transfer-Matrix Approach to Classical Systems" Springer Lecture Note in Physics 528 ed. I. Peschel, X. Wang, K. Hallberg, Springer Berlin (1999) pp. 127-148.

[8] T. Nishino, K. Okunishi, J. Phys. Soc. Jpn. 65, (1996) 891. T. Nishino, K. Okunishi, J. Phys. Soc. Jpn. 66, (1997) 3040. T. Nishino, K. Okunishi, M. Kikuchi, Physics Letters A 213, (1996) 69.

[9] M. Levin, C.P. Nave, Phys. Rev. Lett. 99, 120601 (2007).

[10] T. Nishino, K. Okunishi, J. Phys. Soc. Jpn. 64 (1995) 4084-4087. K. Ueda, T. Nishino, K. Okunishi, Y. Hieida, R. Derian, A. Gendiar, J. Phys. Soc. Jpn. 75, 014003.1-014003.8 (2006).

[11] I. P. McCulloch, arXiv:0804.2509.

[12] G. Vidal, Phys. Rev. Lett. 98, 070201 (2007)

[13] M. Zwolak and G. Vidal, Phys. Rev. Lett. 93, 207205 (2004).

[14] F. Verstraete, J. J. Garcia-Ripoll, J. I. Cirac, Phys. Rev. Lett. 93, 207204 (2004).

[15] Given a matrix M, we refer to its eigenvalue with largest absolute value as the *dominant eigenvalue*. Similarly, we refer to the corresponding eigenvector as the *dominant eigenvector* of M.

[16] J. Jordan, R. Orús, G. Vidal, F. Verstraete, J. I. Cirac, cond-mat/0703788;

[17] Y.-Y. Shi, L.-M. Duan and G. Vidal, Phys. Rev. A 74, 022320 (2006).

[18] The canonical form is unique up to a choice of phases $e^{i\phi_\alpha}$.

Two canonical forms for $|\Psi\rangle$, (Γ, λ) and (Γ', λ'), are related by $(\Gamma')^i_{\alpha\beta} = e^{i\phi_\alpha} \Gamma^i_{\alpha\beta} e^{-i\phi_\beta}$ and $\lambda' = \lambda$.

[19] There are states of a chain, such as the cat state $\lim_{N\to\infty} c_0 |0\rangle^{\otimes N} + c_1 |1\rangle^{\otimes N}$, $|c_0|^2 + |c_1|^2 = 1$, for which the dominant eigenvalue is degenerate. The iMPS description needs to be supplemented with an extra tensor, sitting at infinite, that determines the boundary conditions (in this case the values of c_0 and c_1). Here we will not consider such cases.

[20] Non-Hermitian eigenvalue problems also occur in the context of transfer matrix DMRG. For instance, see N. Shibata, J. Phys. A: Math. and Gen. vol. 36 (2003) R381.

[21] A Cholesky decomposition can also be used to obtain two lower triangular matrices for X and Y (see e.g. *http://en.wikipedia.org/wiki/Cholesky_decomposition*).

[22] This bound is optimal as long as $\chi > \kappa$. In the case $\kappa \geq \chi$ the efficiency can be improved by using alternative contractions.

[23] Truncating a bond index so as to retain the χ largest Schmidt coefficients λ_α is optimal in that it maximizes the overlap between the initial and truncated states. In the present case we use this recipe to truncate all bond indices of the iMPS at once. This is no longer expected to be optimal, but it is simple and seen to produce very satisfactory results.

[24] Another way to turn an iMPS $\{\Gamma^A, \lambda^A, \Gamma^B, \lambda^B\}$ into the canonical form is by using the algorithm of Ref. [12] to simulate a large sequence of trivial two-site gates (that is, gates that implement the identity operator) alternatively acting on even and odd bonds. It is seen that after each update the iMPS is closer to the canonical form. In practice, the orthonormalization strategy explained in this paper is more efficient and precise.

[25] For a non-symmetric Hamiltonian K_2, Q is decomposed into two different matrices $Q = Q_1 Q_2$ (e.g. through a singular value decomposition). If K_2 changes along different lattice directions (anisotropic model), then we will decompose two matrices Q^x and Q^y. In both situations one can proceed in a similar way as in the symmetric, isotropic case.

[26] Our construction was inspired by a similar one in F. Verstraete, M. M. Wolf, D. Perez-Garcia, J. I. Cirac, Phys. Rev. Lett. 96, 220601 (2006), where finite systems were analyzed by mapping the partition function into a PEPS. Here we skip the map into PEPS and significantly reduce simulation costs by decreasing the bond dimension of the resulting 2D tensor network from d^2 to d.

[27] Another good reason to use the canonical form of an iMPS is that it simplifies the comparison between two states. As a criterion for convergence of the sequence $|\Psi_p\rangle$ in Eq. (19), we require that the Schmidt coefficients λ of the iMPS have converged with respect to p within some accuracy.