

The matrix product approximation for the dynamic cavity method

Thomas Barthel

Department of Physics, Duke University, Durham, NC 27708,
United States of America

Received 6 April 2019

Accepted for publication 29 October 2019

Published 23 January 2020



Online at stacks.iop.org/JSTAT/2020/013217

<https://doi.org/10.1088/1742-5468/ab5701>

Abstract. Stochastic dynamics of classical degrees of freedom, defined on vertices of locally tree-like graphs, can be studied in the framework of the dynamic cavity method which is exact for tree graphs. Such models correspond for example to spin-glass systems, Boolean networks, neural networks, and other technical, biological, and social networks. The central objects in the cavity method are edge messages—conditional probabilities of two vertex variable trajectories. In this paper, we discuss a rather pedagogical derivation for the dynamic cavity method, give a detailed account of the novel matrix product edge message (MPEM) algorithm for the solution of the dynamic cavity equation as introduced in Barthel *et al* (2018 *Phys. Rev. E* **97** 010104 (R)), and present optimizations and extensions. Matrix product approximations of the edge messages are constructed recursively in an iteration over time. Computation costs and precision can be tuned by controlling the matrix dimensions of the MPEM in truncations. Without truncations, the dynamics is exact. Data for Glauber–Ising dynamics shows a linear growth of computation costs in time. In contrast to Monte Carlo simulations, the approach has a much better error scaling. Hence, it gives for example access to low probability events and decaying observables like temporal correlations. We discuss optimized truncation schemes and an extension that allows to capture models which have a continuum time limit.

Keywords: stochastic processes, tensor network simulations, dynamic cavity method, message-passing algorithms, network dynamics

Contents

| | |
|--|-----------|
| 1. Introduction | 2 |
| 2. Stochastic dynamics on locally tree-like graphs | 4 |
| 2.1. Global equation of motion | 4 |
| 2.2. Probability for global trajectories | 5 |
| 2.3. Motivation and definition of edge messages | 5 |
| 2.4. Simplification of edge messages | 6 |
| 2.5. Computing edge trajectory probabilities from edge messages | 8 |
| 2.6. Dynamic cavity equation | 8 |
| 3. Matrix product edge messages (MPEM) | 9 |
| 3.1. General idea and motivation | 9 |
| 3.2. Canonical form | 10 |
| 3.3. Controlled truncations | 10 |
| 4. Algorithm for the time evolution of matrix product edge messages | 12 |
| 4.1. One exact MPEM evolution step | 12 |
| 4.2. Truncating MPEMs and recovering the canonical form | 13 |
| 5. Evaluation of observables | 16 |
| 6. An improved truncation scheme | 17 |
| 7. Computation costs in Glauber–Ising dynamics | 20 |
| 8. Models with vertex-state dependence | 21 |
| 9. Discussion | 23 |
| Acknowledgments | 23 |
| References | 23 |

1. Introduction

Statistical physics provides a comprehensive framework for the study of many-body systems in equilibrium with their environment. It is a foundation of modern physics, tracing back the laws of thermodynamics to characteristics of the microscopic degrees of freedom. Beyond physics, its formalism is successfully applied in many different fields such as information theory, computer science, economy, and sociology. In contrast, our methodological toolbox for dynamics in probabilistic systems is rather limited, which restricts our understanding of stochastic dynamical phenomena.

Stochastic dynamics, i.e. dynamics that are governed by probabilistic instead of deterministic rules, are ubiquitous in nature as well as in social and technological systems [1–4]. Here, we focus in particular on stochastic dynamics in networks [5, 6]. Some

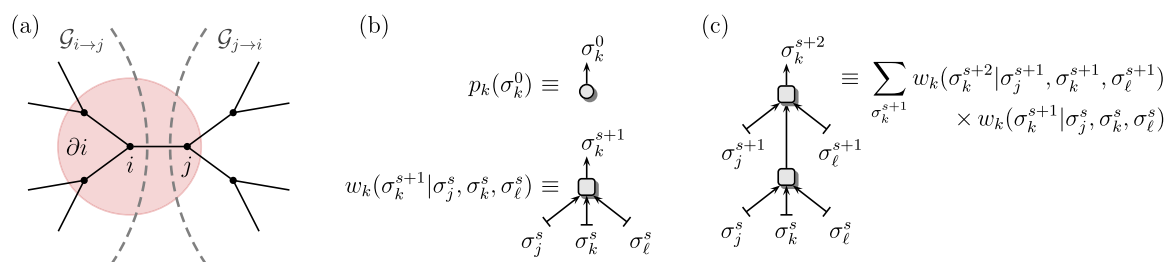


Figure 1. (a) Example for subgraphs $\mathcal{G}_{i \rightarrow j}$ and $\mathcal{G}_{j \rightarrow i}$ as defined in section 2.3. (b) Graphical representations for single-vertex probabilities $p_k(\sigma_k)$ and local transition matrices $w_k(\sigma_k^{s+1} | \sigma_{\partial k}^s)$ for the case of vertex degree 3 ($\partial k = \{j, k, \ell\}$). (c) Graphical representation for the (partial) contraction of two transition matrices.

challenging examples are the thermal dynamics in spin glasses, avalanche dynamics, the dynamics of infectious diseases, the evolution of opinions in social networks, the stability of functionality under perturbations in technological networks, or synchronization phenomena. The probabilistic features can be intrinsic or due to our ignorance of certain details that are not essential for the observed (macroscopic) phenomena. A simple method for the simulation of a stochastic system with states σ_i on N vertices of a network is to propagate the probability distribution $P(\sigma_1^t, \dots, \sigma_N^t)$ in time. Unfortunately, the corresponding computation costs are exponential in the system size N and it is also difficult to assess temporal correlations in this way.

For the investigation of network systems, a strong simplification can be achieved when cycles in the interaction graph are rare or sufficiently long. This is the case for locally tree-like graphs (see figure 1(a)) such as random regular graphs, Erdős–Rényi graphs, and Gilbert graphs. For such random graphs with N vertices, almost all cycles have length $\gtrsim \log N$ such that their effect is negligible in the thermodynamic limit [7]. For static problems, this feature is used in the *cavity method* [8, 9], where conditional nearest-neighbor probabilities are computed iteratively within the Bethe–Peierls approximation. This efficient method has been applied very successfully to study, for example, equilibrium properties of spin glasses [9], computationally hard satisfiability problems [10, 11], and random matrix ensembles [12].

Subsequently, the cavity method has been generalized to dynamical problems, resulting in the *dynamic cavity method* or dynamic belief propagation [13, 14]. The central objects in this approach are not the time-dependent probabilities for global system states, but conditional probabilities $\mu_{i \rightarrow j}(\sigma_i^0, \dots, \sigma_i^t | \sigma_j^0, \dots, \sigma_j^{t-1})$ for state trajectories on neighboring sites i and j (edges $i \rightarrow j$). These so-called edge messages $\mu_{i \rightarrow j}$ are generated in an iteration over time. Consequently, the computational complexity is now linear instead of exponential in the system size N . Unfortunately, the number of possible trajectories and, hence, the computational complexity increase exponentially in time. Applications have thus been restricted to either very short times [13, 15], oriented graphs [13], a no-backtracking approximation [16, 17], or unidirectional dynamics with local absorbing states [14, 18–23]. In the latter case, one can exploit that vertex trajectories can be parametrized by a few switching times.

For general stochastic network dynamics, it is a challenging endeavor to find good approximative solutions to the dynamic cavity equations with polynomial computation costs. A drastic simplification is to neglect temporal correlations completely as in the

one-time approximation (a.k.a. time factorization) [13, 24–26] or to retain only short-time correlations as in the one-step Markov ansatz [27]. While this can be expected to work well for stationary states at high temperatures, such approximations are usually severe for short to intermediate times or low temperatures. Other approximative approaches are the cluster variational method [28–30] (applicable for short-range spatio-temporal correlations) or perturbative schemes [25, 31, 32], generating functional analysis [33–37], and the generalized mean field approximation [38].

In [39], we have introduced a novel efficient algorithm for precise solutions of the parallel dynamic cavity equations for locally tree-like graphs and general bidirectional dynamics. The approach is based on *matrix-product approximations* for the edge messages $\mu_{i \rightarrow j}$. The computational complexity is decreased from exponential to polynomial in the duration of the dynamical process. As we demonstrated for non-equilibrium Glauber dynamics in the kinetic Ising model, one can obtain quasi-exact results with relatively small matrix dimensions. Computation costs and accuracy can be tuned through controlled truncations of the matrix dimensions. Matrix-product approximations are widely used in condensed matter theory to encode spatial correlations of quantum states [40–43]. Here, we employ matrix products to encode temporal correlations in the edge messages.

This paper gives a pedagogical introduction to the dynamic cavity method in section 2. Sections 3–5 give a detailed account of the novel matrix product edge message (MPEM) algorithm [39] for the solution of the dynamic cavity equation. Section 6 provides an improved MPEM truncation scheme that substantially reduces computation costs. For Glauber–Ising dynamics, we demonstrate a linear growth of computation costs in section 7. In section 8, the MPEM method is generalized to systems where the transition probability for the state of a vertex does not only depend on the state of its neighbors in the interaction graph, but also on the current state of the vertex itself.

2. Stochastic dynamics on locally tree-like graphs

2.1. Global equation of motion

Let us consider a graph \mathcal{G} and denote the state at vertex $i \in \mathcal{G}$ by σ_i . The state of the full system at time t will be denoted by

$$\boldsymbol{\sigma}^t := (\sigma_1^t, \sigma_2^t, \dots).$$

Given the state probabilities $P(\boldsymbol{\sigma}^t)$ for time t , the probabilities $P(\boldsymbol{\sigma}^{t+1})$ for the subsequent time step $t+1$ are obtained by applying the transition matrix $W(\boldsymbol{\sigma}^{t+1}|\boldsymbol{\sigma}^t)$, where the notation of the arguments indicates that W is a conditional probability.

$$P(\boldsymbol{\sigma}^{t+1}) = \sum_{\boldsymbol{\sigma}^t} W(\boldsymbol{\sigma}^{t+1}|\boldsymbol{\sigma}^t) P(\boldsymbol{\sigma}^t) \quad \text{with} \quad \sum_{\boldsymbol{\sigma}} W(\boldsymbol{\sigma}|\boldsymbol{\sigma}') = 1. \quad (1)$$

In correspondence with the structure of the graph \mathcal{G} , we assume that, in every time step, the probability for σ_i^{t+1} only depends on the states σ_k^t of nearest-neighbors k at the previous time step such that the global transition matrix W is a product of local transition matrices w_i ,

$$W(\sigma^{t+1}|\sigma^t) = \prod_{i \in \mathcal{G}} w_i(\sigma_i^{t+1}|\sigma_{\partial i}^t) \quad \text{with} \quad \sum_{\sigma_i} w_i(\sigma_i|\sigma'_{\partial i}) = 1, \quad (2)$$

where the vicinity $\partial i := \{k \in \mathcal{G} \mid \text{dist}(i, k) \leq 1\}$ of vertex i , contains i and its nearest neighbors. Equations (1) and (2) specify the parallel stochastic dynamics of the system.

2.2. Probability for global trajectories

With the definitions above, the probability for a trajectory $\sigma^0 \rightarrow \sigma^1 \rightarrow \dots \rightarrow \sigma^t$ of states up to time t takes the form

$$P(\sigma^0 \sigma^1 \dots \sigma^{t-1} \sigma^t) = \prod_{s=1}^t W(\sigma^s|\sigma^{s-1}) P(\sigma^0). \quad (3)$$

For simplicity, let us assume initially uncorrelated states, i.e.

$$P(\sigma^0) = \prod_{i \in \mathcal{G}} p_i(\sigma_i^0) \quad \text{with} \quad \sum_{\sigma_i} p_i(\sigma_i) = 1. \quad (4)$$

2.3. Motivation and definition of edge messages

An exact solution of the global equation of motion (1) is usually not possible. In principle, one can employ a simple iteration over time. However, the corresponding memory costs (to keep track of $P(\sigma^t)$) and computation costs scale *exponentially* in the system size $|\mathcal{G}|$ such that this approach is limited to very small systems.

A very useful technique is the Markov chain Monte Carlo method. In this approach, one repeatedly generates states σ^0 according to the $t=0$ probability $P(\sigma^0)$, and propagates them in time by choosing a state σ^{t+1} with probability $W(\sigma^{t+1}|\sigma^t)$ for $t=0, 1, 2, \dots$. While the technique can be used to study many interesting problems, it has several disadvantages. First of all, one cannot address the thermodynamic limit directly and needs to perform a costly finite-size scaling analysis. Secondly, the number of required samples can be very large if temporal correlations are non-trivial. Also, the error scaling is not very favorable. When increasing the number of samples N_s , errors decrease slowly as $1/\sqrt{N_s}$. Especially if the absolute value of the desired observable is small, as is usually the case when studying temporal correlations, it is often impossible to achieve a sufficient precision.

To resolve these issues in our approach for locally tree-like graphs, we shift the focus from global-state probabilities $P(\sigma^t)$ to so-called edge messages. These are conditional probabilities for the *trajectory* $\bar{\sigma}_i^{t+1}$ of the variable at vertex i ,

$$\bar{\sigma}_i^{t+1} := (\sigma_i^0, \sigma_i^1, \dots, \sigma_i^t, \sigma_i^{t+1}), \quad (5)$$

and the trajectory $\bar{\sigma}_j^t$ on a neighboring site j . If the graph is a tree, the exact evolution of the system can be formulated in terms of the edge messages and leads to the dynamic cavity equations [13, 14] discussed in the following. If the graph contains some longer loops, i.e. is only locally tree-like, the dynamic cavity equations give an approximation to the exact dynamics with the accuracy depending on the number and lengths of loops.

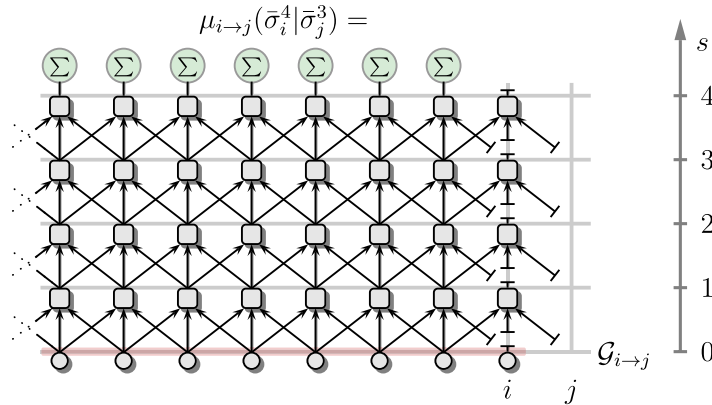


Figure 2. Edge message $\mu_{i \rightarrow j}(\bar{\sigma}_i^4 | \bar{\sigma}_j^3)$, as defined in equation (6), for a one-dimensional graph.

When we remove an edge (i, j) from a tree graph \mathcal{G} , it is decomposed into two parts $\mathcal{G}_{i \rightarrow j}$ and $\mathcal{G}_{j \rightarrow i}$ such that $\mathcal{G}_{i \rightarrow j}$ contains i and the subgraph that is still connected to i (and all subgraphs of \mathcal{G} that are disconnected from i and j) while $\mathcal{G}_{j \rightarrow i}$ contains j and the subgraph that is still connected to j . See figure 1(a). Now, an *edge message* $\mu_{i \rightarrow j}(\bar{\sigma}_i^{t+1} | \bar{\sigma}_j^t)$ for edge (i, j) is the probability of the trajectory $\bar{\sigma}_i^{t+1}$ on vertex i for the dynamics being restricted to the subgraph $\mathcal{G}_{i \rightarrow j}$ under the condition that we impose the trajectory $\bar{\sigma}_j^t$ on vertex j . Specifically, it is the product of all transition matrices $w_k(\sigma_k^{s+1} | \sigma_{\partial k}^s)$ and $p_k(\sigma_k^0)$ for all $k \in \mathcal{G}_{i \rightarrow j}$ and $0 \leq s \leq t$, summed over all $\bar{\sigma}_k^{t+1}$ with $k \in \mathcal{G}_{i \rightarrow j} \setminus \{i\}$, i.e.

$$\mu_{i \rightarrow j}(\bar{\sigma}_i^{t+1} | \bar{\sigma}_j^t) := \sum_{\{\bar{\sigma}_k^{t+1}\}_{k \in \mathcal{G}_{i \rightarrow j} \setminus \{i\}}} \left[\prod_{s=0}^t \prod_{q \in \mathcal{G}_{i \rightarrow j}} w_q(\sigma_q^{s+1} | \sigma_{\partial q}^s) \right] \prod_{q \in \mathcal{G}_{i \rightarrow j}} p_q(\sigma_q^0). \quad (6)$$

A pictorial representation for a one-dimensional graph is given in figure 2.

2.4. Simplification of edge messages

Due to the normalization constraint $\sum_{\sigma_i} w_i(\sigma_i | \sigma'_{\partial i}) = 1$ of the transition matrices, the expression (6) for the edge message can be simplified substantially by removing all w_k and p_k outside the *causal cone* of edge $i \rightarrow j$ such that

$$\mu_{i \rightarrow j}(\bar{\sigma}_i^{t+1} | \bar{\sigma}_j^t) = \sum_{\{\sigma_k^s\}_{(k,s) \in \bar{\mathcal{G}}_{i \rightarrow j}^{t+1}}} \left[\prod_{s=0}^t \prod_{q \in \mathcal{G}_{i \rightarrow j}^{t-s}} w_q(\sigma_q^{s+1} | \sigma_{\partial q}^s) \right] \prod_{q \in \mathcal{G}_{i \rightarrow j}^{t+1}} p_q(\sigma_q^0). \quad (7)$$

In this equation, we use the (recursively defined) subgraphs

$$\mathcal{G}_{i \rightarrow j}^0 := \{i\}, \quad \mathcal{G}_{i \rightarrow j}^{s+1} := \mathcal{G}_{i \rightarrow j}^s \cup \left[\bigcup_{k \in \mathcal{G}_{i \rightarrow j}^s} (\partial k \setminus \{j\}) \right] = \{k \in \mathcal{G}_{i \rightarrow j} \mid \text{dist}(i, k) \leq s+1\} \quad (8)$$

and the *causal cone* $\bar{\mathcal{G}}_{i \rightarrow j}^{t+1}$ of edge $i \rightarrow j$ which is a subset of $\mathbb{N}_0 \times \mathcal{G}_{i \rightarrow j}$,

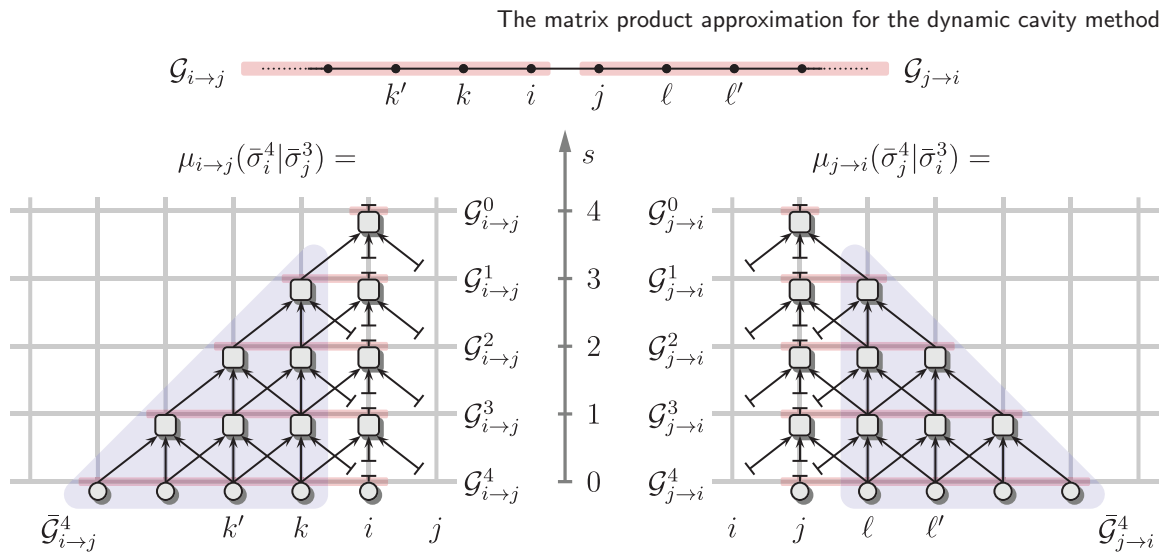


Figure 3. Simplified forms of edge messages $\mu_{i \rightarrow j}(\bar{\sigma}_i^4 | \bar{\sigma}_j^3)$ and $\mu_{j \rightarrow i}(\bar{\sigma}_j^4 | \bar{\sigma}_i^3)$ for a one-dimensional graph, where, as in equation (7), the normalization constraints of transition matrices w_k and $t=0$ probabilities $p_k(\sigma_k^0)$ have been used to remove those outside the causal cones $\bar{\mathcal{G}}_{i \rightarrow j}^4$ and $\bar{\mathcal{G}}_{j \rightarrow i}^4$. The latter are defined in equation (9) and are built from subgraphs $\mathcal{G}_{i \rightarrow j}^s$ and $\mathcal{G}_{j \rightarrow i}^s$, respectively.

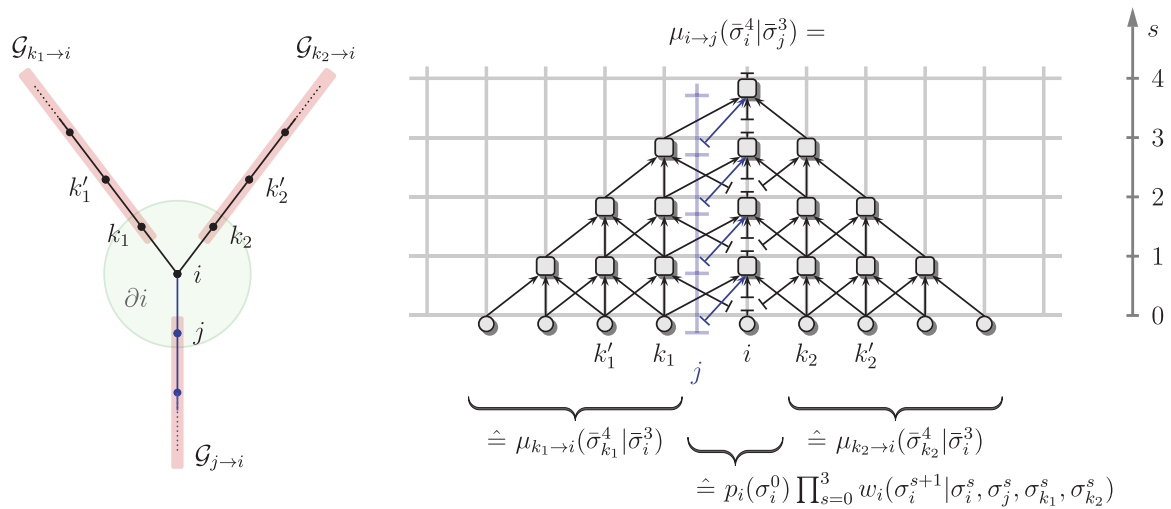


Figure 4. Simplified form (7) of the edge message $\mu_{i \rightarrow j}(\bar{\sigma}_i^4 | \bar{\sigma}_j^3)$ for a Y-junction graph with central vertex i , where normalization constraints have been used to remove transition matrices w_k and $t=0$ probabilities $p_k(\sigma_k^0)$ outside the causal cone $\bar{\mathcal{G}}_{i \rightarrow j}^4$. As shown here and specified in equation (11), the edge message $\mu_{i \rightarrow j}$ for time $t+1$ can be obtained by contracting $\mu_{k \rightarrow i}$ for time t and all vertices $k \neq i, j$ in the neighborhood ∂i with transition matrices $w_i(\sigma_i^{s+1} | \sigma_{\partial i}^s)$ and $p_i(\sigma_i^0)$.

$$\bar{\mathcal{G}}_{i \rightarrow j}^t := \bigcup_{s=0}^t \left\{ (s, k) \mid k \in \mathcal{G}_{i \rightarrow j}^{t-s} \setminus \{i\} \right\}. \quad (9)$$

See figures 3 and 4 for examples.

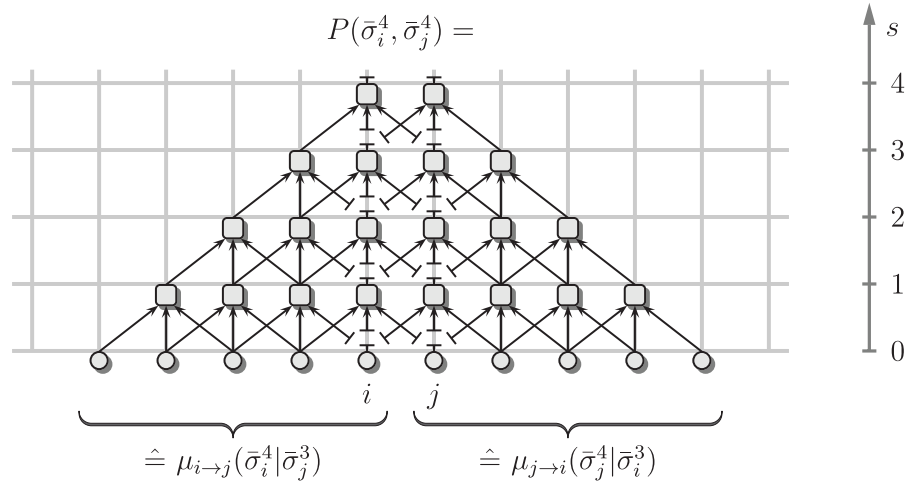


Figure 5. The joint probability (10) for the trajectories $\bar{\sigma}_i^4$ and $\bar{\sigma}_j^4$ on an edge (i, j) is given by the product of the edge messages $\mu_{i \rightarrow j}(\bar{\sigma}_i^4 | \bar{\sigma}_j^3)$ and $\mu_{j \rightarrow i}(\bar{\sigma}_j^4 | \bar{\sigma}_i^3)$. It is shown here for a one-dimensional graph.

2.5. Computing edge trajectory probabilities from edge messages

Before showing that edge messages can be generated in an iteration (section 2.6), note that they allow us to evaluate easily many observables of interest. In particular, the product of edge messages $\mu_{i \rightarrow j}(\bar{\sigma}_i^t | \bar{\sigma}_j^{t-1})$ and $\mu_{j \rightarrow i}(\bar{\sigma}_j^t | \bar{\sigma}_i^{t-1})$ yields the joint probability for the occurrence of trajectories $\bar{\sigma}_i^t$ and $\bar{\sigma}_j^t$ on vertices i and j ,

$$P(\bar{\sigma}_i^t, \bar{\sigma}_j^t) \equiv \sum_{\{\bar{\sigma}_k^t\}_{k \in \mathcal{G} \setminus \{i, j\}}} P(\sigma^0 \sigma^1 \dots \sigma^{t-1} \sigma^t) = \mu_{i \rightarrow j}(\bar{\sigma}_i^t | \bar{\sigma}_j^{t-1}) \mu_{j \rightarrow i}(\bar{\sigma}_j^t | \bar{\sigma}_i^{t-1}). \quad (10)$$

See figure 5.

By marginalizing equation (10) over certain subsets of the variables $\{\sigma_i^s\}_{0 \leq s \leq t}$ and $\{\sigma_j^s\}_{0 \leq s \leq t}$, one obtains the probabilities required for the evaluation of time-local observables or temporal correlation functions.

2.6. Dynamic cavity equation

Inspection of the simplified expression (7) for edge message $\mu_{i \rightarrow j}(\bar{\sigma}_i^{t+1} | \bar{\sigma}_j^t)$ shows that it can be constructed from the time- t edge messages $\mu_{k \rightarrow i}(\bar{\sigma}_k^t | \bar{\sigma}_i^{t-1})$ according to

$$\mu_{i \rightarrow j}(\bar{\sigma}_i^{t+1} | \bar{\sigma}_j^t) = \sum_{\{\bar{\sigma}_k^t\}_{k \in \partial i \setminus \{i, j\}}} p_i(\sigma_i^0) \left[\prod_{s=0}^t w_i(\sigma_i^{s+1} | \sigma_{\partial i}^s) \right] \left[\prod_{k \in \partial i \setminus \{i, j\}} \mu_{k \rightarrow i}(\bar{\sigma}_k^t | \bar{\sigma}_i^{t-1}) \right] \quad (11)$$

with $\mu_{i \rightarrow j}(\sigma_i^0) \equiv p_i(\sigma_i^0)$. This recursion relation for the edge messages, the *dynamic cavity equation* [13, 14], is very useful and central for the matrix product edge message time-evolution algorithm presented in section 4. An illustration of equation (11) for the example of a Y-junction graph is given in figure 4.

3. Matrix product edge messages (MPEM)

This section and section 4 give a detailed account of an evolution algorithm that employs precise approximations to edge messages (7) in matrix product form [39]. Extensions will be described in sections 6 and 8.

3.1. General idea and motivation

In the following let d denote the number of different vertex states ($\sigma_i = 1, \dots, d$). In section 2, we have seen that the stochastic dynamics on a tree graph \mathcal{G} can not only be simulated by sampling trajectories ($\sigma^0 \sigma^1 \dots \sigma^{t-1} \sigma^t$) of the full system according to their weights (1) using a Monte Carlo algorithm, but also by an iterative construction of edge messages $\mu_{i \rightarrow j}(\bar{\sigma}_i^t | \bar{\sigma}_j^{t-1})$ which give access to many observables of interest such as the evolution of time-local observables or temporal correlators. Importantly, the computation costs and memory requirements for evolving and storing an edge message ($\mathcal{O}(d^{2t-1})$) are independent of the system size. However, without approximations, the costs increase exponentially with time t . So, an exact treatment is limited to short times.

To resolve this problem, we can employ an idea from the *density matrix renormalization group* (DMRG) [44–46] which is a numerical algorithm for the simulation of strongly-correlated quantum many-body systems (predominantly for one-dimensional lattices) which has been used very successfully for a wide range of applications in condensed matter physics, the physics of ultracold atomic gases, and quantum chemistry. In principle, the encoding of a many-body state for a lattice of L sites,

$$|\psi\rangle = \sum_{n_1, \dots, n_L} \psi_{n_1, \dots, n_L} |\mathbf{n}\rangle, \quad \text{where} \quad |\mathbf{n}\rangle \equiv |n_1\rangle \otimes |n_2\rangle \otimes \dots \otimes |n_L\rangle \quad (12)$$

label orthonormal basis states ($\langle \mathbf{n} | \mathbf{n}' \rangle = \delta_{\mathbf{n}\mathbf{n}'}$), requires storage of the d^L expansion coefficients ψ_{n_1, \dots, n_L} . However, in ground states of typical quantum many-body systems, spatial correlations like $\langle \hat{S}_x^+ \hat{S}_{x'}^- \rangle$ decay quickly in the distance $|x - x'|$, exponentially or algebraically. Due to this fact and corresponding entanglement properties, one can approximate ψ_{n_1, \dots, n_L} by much fewer effective degrees of freedom, given by the elements of $M_x \times M_{x+1}$ matrices $A_x^{n_x}$ in the approximation [47–50]

$$|\psi\rangle \approx \sum_{\mathbf{n}} A_1^{n_1} A_2^{n_1} \dots A_{L-1}^{n_{L-1}} A_L^{n_L} |\mathbf{n}\rangle. \quad (13)$$

The M_x are called *bond dimensions*. In order for the matrix product $A_1^{n_1} \dots A_L^{n_L}$ to yield a scalar, we require dimensions $M_1 = M_{L+1} = 1$ at the system boundaries. The elements of matrices $A_x^{n_x}$ are effective degrees of freedom that encode correlations around site x . The larger the bond dimensions M_x are chosen, the more effective degrees of freedom are taken into account, the higher the computation costs, and the higher the precision of the approximation. In any case, the memory costs ($\mathcal{O}(dLM^2)$) and computation costs ($\mathcal{O}(d^k LM^3)$) are now linear in the system size L . The right-hand side of equation (13) is called a *matrix product state* (MPS) [40–43]. More recently, MPS are also discussed in the more mathematical literature, sometimes under the name *tensor train decomposition* [51]. DMRG is a class of algorithms operating on MPS—most importantly,

The matrix product approximation for the dynamic cavity method

Figure 6. Canonical form (14) of an MPEM $\mu_{i \rightarrow j}(\bar{\sigma}_i^t | \bar{\sigma}_j^{t-1})$. For every time step s with $1 \leq s \leq t-1$, there are d^2 matrices $A_{i \rightarrow j}^{(s)}(\sigma_i^{s-1} | \sigma_j^s)$ of dimension $M_s \times M_{s+1}$. With $M := \max_s M_s$, the number of degrees of freedom is of order td^2M^2 , i.e. linear in time instead of exponential in time.

to compute ground state approximations, thermal equilibrium states, or to study non-equilibrium phenomena. Observables can often be computed to machine precision.

The idea is now to similarly exploit that (connected) *temporal* correlations in edge messages often decay quickly in time t and/or in the time difference $|t - t'|$ and to approximate edge messages $\mu_{i \rightarrow j}(\bar{\sigma}_i^t | \bar{\sigma}_j^{t-1})$ in the form of a matrix product.

3.2. Canonical form

Let us define the canonical form of a *matrix product edge message* (MPEM) as

$$\mu_{i \rightarrow j}(\bar{\sigma}_i^t | \bar{\sigma}_j^{t-1}) = A_{i \rightarrow j}^{(0)}(\sigma_j^0) \left[\prod_{s=1}^{t-1} A_{i \rightarrow j}^{(s)}(\sigma_i^{s-1} | \sigma_j^s) \right] A_{i \rightarrow j}^{(t)}(\sigma_i^{t-1}) A_{i \rightarrow j}^{(t+1)}(\sigma_i^t). \quad (14)$$

The graphical representation is shown in figure 6. The particular choice of assigning state labels $\{\sigma_i^s\}$ and $\{\sigma_j^s\}$ to the $M_s \times M_{s+1}$ matrices occurring in the matrix product (14) is advantageous for the implementation of the dynamic cavity equation (11) for MPEMs as will become clear in section 4. In order for the matrix product (14) to yield a scalar, we set $M_0 = M_{t+2} = 1$.

3.3. Controlled truncations

When advancing an MPEM $\mu_{i \rightarrow j}$ one step in time according to the dynamic cavity equation (11), we need to contract¹ several MPEMs, $\mu_{k \rightarrow i}(\bar{\sigma}_i^t | \bar{\sigma}_j^{t-1})$ with k from the neighborhood ∂i of vertex i , and local transition matrices w_i and, subsequently, express the result again in MPEM form. In general, this generates an MPEM $\mu_{i \rightarrow j}(\bar{\sigma}_i^{t+1} | \bar{\sigma}_j^t)$ with increased bond dimension $M' \geq M$. In order to control the growth of the bond dimensions, and hence the computation costs, we would like to reduce the bond dimension of the new MPEM. This can indeed be done in a controlled fashion such that the resulting truncated MPEM is close to the original. Here, we discuss the most simple truncation scheme [39] and an optimized scheme is presented in section 6. The iterative time evolution of MPEMs is described in section 4.

Because the notation is a little simpler, let us demonstrate truncations using the example of an MPS for a quantum system of L sites,

$$|\psi\rangle = \sum_{\mathbf{n}} A_1^{n_1} A_2^{n_1} \cdots A_{L-1}^{n_{L-1}} A_L^{n_L} |\mathbf{n}\rangle. \quad (15)$$

¹ With the *contraction* of two tensors, we refer to their product, summed over certain sets of common indices.

Let us split the system into two parts \mathcal{L} , containing sites $[1, \ell]$, and \mathcal{R} , containing sites $[\ell + 1, L]$. Let $\{|a\rangle\}$ and $\{|b\rangle\}$ be orthonormal bases for left and right parts, respectively, such that

$$|\psi\rangle =: \sum_{a,b} \psi_{a,b} |a\rangle \otimes |b\rangle. \quad (16)$$

The objective is to find a good approximation $|\psi_{\text{trunc}}\rangle$ of $|\psi\rangle$ in a reduced vector space, where the 2-norm distance is used to quantify the accuracy. This can be done using a singular value decomposition (SVD) of $\psi_{a,b}$. In matrix form, it reads $\psi = U\Lambda V^\dagger$, where U and V are unitary matrices and Λ is the diagonal matrix containing descendingly ordered *singular values* $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_{M'}$ such that

$$|\psi\rangle = \sum_{a,b,k} U_{a,k} \lambda_k V_{b,k}^* |a\rangle \otimes |b\rangle = \sum_{k=1}^{M'} \lambda_k |k_{\mathcal{L}}\rangle \otimes |k_{\mathcal{R}}\rangle, \quad (17a)$$

where the states $\{|k_{\mathcal{L}}\rangle := \sum_a U_{a,k} |a\rangle\}$ and $\{|k_{\mathcal{R}}\rangle := \sum_b V_{b,k}^* |b\rangle\}$ are also orthonormal bases for \mathcal{L} and \mathcal{R} , respectively. The best approximation $|\psi\rangle$ with $M < M'$ nonzero singular values (also known as Schmidt coefficients in this case) is given by

$$|\psi_{\text{trunc}}\rangle := \sum_{k=1}^M \lambda_k |k_{\mathcal{L}}\rangle \otimes |k_{\mathcal{R}}\rangle \quad \text{with error} \quad \|\psi - \psi_{\text{trunc}}\|^2 = \sum_{k=M+1}^{M'} \lambda_k^2. \quad (17b)$$

The fact that this procedure yields the best rank- M approximation of $|\psi\rangle$ for the given bipartition corresponds to the Eckart–Young theorem [52, 53].

So, to truncate in this fashion bond dimensions of an MPS $|\psi\rangle$ (and analogously for MPEM), we need to express it in suitable orthonormal bases for \mathcal{L} and \mathcal{R} . This can be achieved by exploiting the freedom to replace two subsequent matrices $(A_x^{n_x}, A_{x+1}^{n_{x+1}})$ in the matrix product (15) by $(A_x^{n_x} X^{-1}, X A_{x+1}^{n_{x+1}})$. The inserted non-singular matrices X and X^{-1} clearly leave $|\psi\rangle$ invariant as they cancel in the matrix product. Using this gauge freedom, we can bring the matrix product (15) into the form

$$|\psi\rangle = \sum_{\mathbf{n}} Y_1^{n_1} \dots Y_\ell^{n_\ell} \cdot C \cdot Z_{\ell+1}^{n_{\ell+1}} \dots Z_L^{n_L} |\mathbf{n}\rangle \quad \text{with} \quad (18a)$$

$$\sum_{\mathbf{n}} (Y_i^n)^\dagger Y_i^n = \mathbb{1} \quad \text{and} \quad \sum_{\mathbf{n}} Z_i^n (Z_i^n)^\dagger = \mathbb{1}. \quad (18b)$$

The left and right *orthonormality constraints* (18b) are imposed by sequences of singular value decompositions, sweeping from site 1 to site ℓ and from site L to site $\ell + 1$ with details discussed in section 4.2. The resulting matrix product (18) is in fact of the form (16), with $\psi_{a,b} \leftrightarrow C_{a,b}$, and orthonormal basis states

$$|a\rangle = \sum_{n_1, \dots, n_\ell} [Y_1^{n_1} \dots Y_\ell^{n_\ell}]_{1,a} |n_1 \dots n_\ell\rangle \quad \text{for } \mathcal{L} \text{ and} \quad (19a)$$

$$|b\rangle = \sum_{n_{\ell+1}, \dots, n_L} [Z_{\ell+1}^{n_{\ell+1}} \dots Z_L^{n_L}]_{b,1} |n_{\ell+1} \dots n_L\rangle \quad \text{for } \mathcal{R}. \quad (19b)$$

The orthonormality of these states follows from equation (18b):

$$\begin{aligned}\langle a'|a\rangle &= \sum_{n_1, \dots, n_\ell} [(Y_\ell^{n_\ell})^\dagger \cdots (Y_3^{n_3})^\dagger (Y_2^{n_2})^\dagger (Y_1^{n_1})^\dagger Y_1^{n_1} Y_2^{n_2} Y_3^{n_3} \cdots Y_\ell^{n_\ell}]_{a', a} \\ &= \sum_{n_2, \dots, n_\ell} [(Y_\ell^{n_\ell})^\dagger \cdots (Y_3^{n_3})^\dagger (Y_2^{n_2})^\dagger Y_2^{n_2} Y_3^{n_3} \cdots Y_\ell^{n_\ell}]_{a', a} = \cdots = [\mathbb{1}]_{a', a} = \delta_{a, a'}\end{aligned}\quad (20)$$

and similarly for the states $\{|b\rangle\}$. With a singular value decomposition of the matrix C , we then obtain an optimally truncated state $|\psi_{\text{trunc}}\rangle$ (equation (17b)) in MPS form.

4. Algorithm for the time evolution of matrix product edge messages

Given MPEMs $\mu_{i \rightarrow j}(\bar{\sigma}_i^t | \bar{\sigma}_j^{t-1})$ (equation (14)) for all edges at time t , we want to do one time step according to the dynamic cavity equation (11) and obtain MPEM approximations

$$\mu_{i \rightarrow j}(\bar{\sigma}_i^{t+1} | \bar{\sigma}_j^t) = B_{i \rightarrow j}^{(0)}(\sigma_j^0) \left[\prod_{s=1}^t B_{i \rightarrow j}^{(s)}(\sigma_i^{s-1} | \sigma_j^s) \right] B_{i \rightarrow j}^{(t+1)}(\sigma_i^t) B_{i \rightarrow j}^{(t+2)}(\sigma_i^{t+1}) \quad (21)$$

for the edge messages at time $t+1$.

Let us assume for now that vertex i is not member of its neighborhood ∂i , i.e. that the local transition matrix $w_i(\sigma_i^{s+1} | \sigma_{\partial i}^s)$ is independent of σ_i^s . The more general case can also be handled as well but requires a slightly more complicated algorithm. The corresponding extension is described in section 8.

4.1. One exact MPEM evolution step

First, we generate a non-canonical matrix product representation of the evolved edge message (11), in particular, choosing the non-canonical form

$$\mu_{i \rightarrow j}(\bar{\sigma}_i^{t+1} | \bar{\sigma}_j^t) = C_{i \rightarrow j}^{(0)}(\sigma_j^0) \left[\prod_{s=1}^{t+1} C_{i \rightarrow j}^{(s)}(\sigma_i^s | \sigma_j^{s-1}) \right]. \quad (22)$$

To this purpose, the matrices $C_{i \rightarrow j}^{(s)}(\sigma_i^s | \sigma_j^{s-1})$ for the bulk, $0 < s < t$, are obtained by contracting the transition matrix $w_i(\sigma_i^s | \sigma_{\partial i}^{s-1})$ for vertex i with the tensors $A_{k \rightarrow i}^{(s)}(\sigma_k^{s-1} | \sigma_i^s)$ from the time- t MPEMs for edges $k \rightarrow i$ with vertices $k \in \partial i \setminus \{j\} =: \{k_1, \dots, k_{z-1}\}$, where z is the degree of vertex i . In this contraction, we sum over the $z-1$ common indices σ_k^{s-1} as illustrated in figure 7. The resulting matrices

$$C_{i \rightarrow j}^{(s)}(\sigma_i^s | \sigma_j^{s-1}) = \sum_{\sigma_{k_1}^{s-1} \cdots \sigma_{k_{z-1}}^{s-1}} w_i(\sigma_i^s | \sigma_{\partial i}^{s-1}) \left[\bigotimes_{n=1}^{z-1} A_{k_n \rightarrow i}^{(s)}(\sigma_{k_n}^{s-1} | \sigma_i^s) \right] \quad (23)$$

have left and right multi-indices of dimensions $\bar{M}_s = (M_s)^{z-1}$ and $\bar{M}_{s+1} = (M_{s+1})^{z-1}$, corresponding to the direct products $\mathbf{a} := (a_{k_1}, \dots, a_{k_{z-1}})$ and $\mathbf{b} := (b_{k_1}, \dots, b_{k_{z-1}})$ of the left and right indices of the matrices $A_{k \rightarrow i}^{(s)}(\sigma_k^{s-1} | \sigma_i^s)$. For notational simplicity, we have

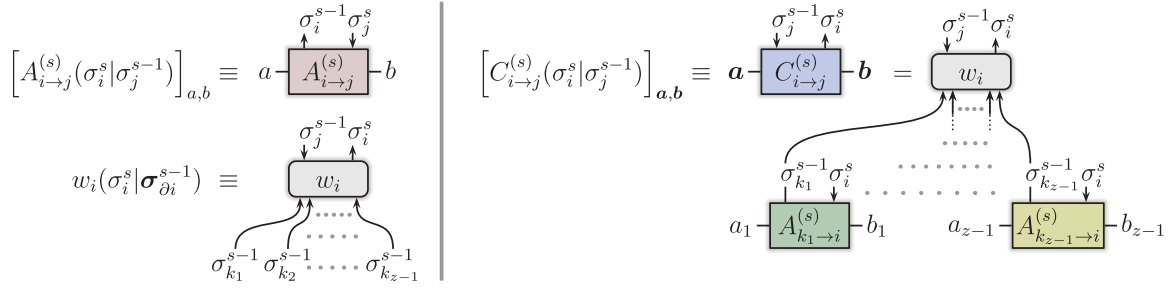


Figure 7. Construction of tensors $C_{i \rightarrow j}^{(s)}$ for the evolved MPEM $\mu_{i \rightarrow j}(\bar{\sigma}_i^{t+1} | \bar{\sigma}_j^t)$ of edge $i \rightarrow j$ at time $t + 1$ using the local transition matrix $w_i(\sigma_i^s | \sigma_{\partial i}^{s-1})$ and tensors $A_{k \rightarrow i}^{(s)}$ of time- t MPEMs of neighboring edges $k \rightarrow i$. The resulting MPEM (22) is not yet in the canonical form (21). The contraction shown here applies to the bulk $0 < s < t$. The contractions at the boundaries $s = 0$ and $s = t, t + 1$ are shown in figure 8.

assumed here that the bond dimensions M_s are the same for all time- t MPEMs of the edges $k \rightarrow i$ with $k \in \{k_1, \dots, k_{z-1}\}$.

Specifically, for the case $z = 3$ with $\partial i = \{j, k_1, k_2\}$, the contractions take for example the form

$$\begin{aligned} [C_{i \rightarrow j}^{(s)}(\sigma_i^s | \sigma_j^{s-1})]_{(a_1, a_2), (b_1, b_2)} &= \sum_{\sigma_{k_1}^{s-1}, \sigma_{k_2}^{s-1}} w_i(\sigma_i^s | \sigma_j^{s-1} \sigma_{k_1}^{s-1} \sigma_{k_2}^{s-1}) \\ &\quad \times [A_{k_1 \rightarrow i}^{(s)}(\sigma_{k_1}^{s-1} | \sigma_i^s)]_{a_1, b_1} \cdot [A_{k_2 \rightarrow i}^{(s)}(\sigma_{k_2}^{s-1} | \sigma_i^s)]_{a_2, b_2}. \end{aligned}$$

At the extremal time slices $s = 0$ and $s = t, t + 1$, the contractions differ slightly. They are specified graphically in figure 8.

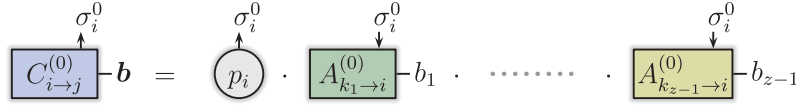
At this point, it has become clear why the specific choice for the assignment of state indices $\{\sigma_i^s | 0 \leq s \leq t\}$ and $\{\sigma_j^s | 0 \leq s \leq t - 1\}$ to tensors $A_{i \rightarrow j}^{(s)}$ in the canonical MPEM form (14) is favorable. It allows for an entirely local construction of tensor $C_{i \rightarrow j}^{(s)}$ from tensors $A_{k \rightarrow i}^{(s)}$, i.e. only A -tensors of a single time-slice s are involved.

4.2. Truncating MPEMs and recovering the canonical form

The exact progression by one time step yields the non-canonical MPEM (22) with bond dimensions being increased from M_s to $\tilde{M}_s = (M_s)^{z-1}$. If we would proceed without any approximation, the computation cost would increase exponentially in time and, hence, restrict the simulation to very short times. Therefore, we are faced with two objectives: (a) reducing bond dimensions by a controlled truncation of the MPEM as described in section 3.3 and motivated physically in section 3.1, and (b) rearranging the assignment of physical state indices $\{\sigma_i^s\}$ and $\{\sigma_j^s\}$ to attain the canonical form (21) of the evolved MPEM. This can be achieved in different ways: a relatively simple method [39] is described in the following and an optimized scheme is introduced in section 6.

If we want to truncate bond s , i.e. reduce the bond dimension \tilde{M}_s to something smaller, we need to take care of orthonormality as discussed in section 3.3. In particular, we need to express the edge message in orthonormal reduced bases for left block $[0, s]$ and right block $[s + 1, t + 1]$. This can be done by imposing left and right

Left boundary ($s = 0$):



Right boundary ($s = t, t + 1$):

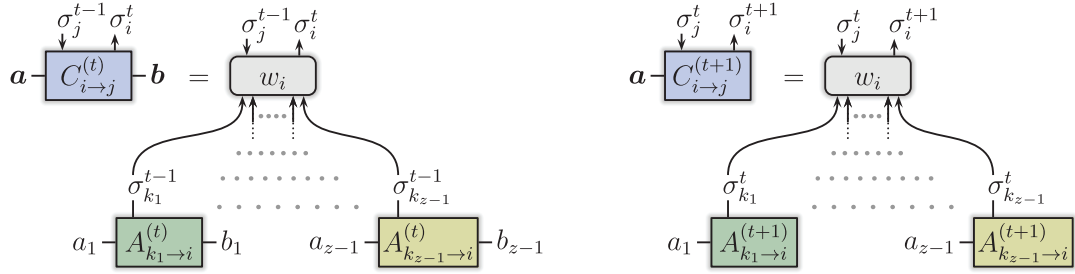


Figure 8. Construction of the evolved MPEM $\mu_{i \rightarrow j}(\bar{\sigma}_i^{t+1} | \bar{\sigma}_j^t)$ in the noncanonical form (22), using transition matrices and MPEM tensors from the previous time step. The contractions shown here concern the boundaries $s = 0$ and $s = t, t + 1$. Those for the bulk $0 < s < t$ are shown in figure 7.

orthonormality constraints (18b) for the C -tensors. If we would not do so and truncate bond dimension \tilde{M}_s , say, through an SVD of tensor $C^{(s)}$ without having the other tensors in orthonormal form, the approximation error would be uncontrolled. Truncating the smallest singular values of $C^{(s)}$ would then not correspond to the best rank- M_s approximation.

In a first sweep from right ($s = t + 1$) to left ($s = 0$), we can sequentially impose the right orthonormality constraint $\sum_n Z_i^n (Z_i^n)^\dagger = \mathbb{1}$ on all C -tensors. As the original C -tensors do not obey the orthonormality constraints, we can only truncate singular values λ_k that are (up to machine precision) zero. At the right boundary, we start with the singular value decomposition $C_{i \rightarrow j}^{(t+1)}(\sigma_i^{t+1} | \sigma_j^t) \stackrel{\text{SVD}}{=} U^{(t+1)} \Lambda^{(t+1)} \tilde{C}_{i \rightarrow j}^{(t+1)}(\sigma_i^{t+1} | \sigma_j^t)$, continue for the bulk tensors $s = t, t - 1, \dots, 1$ with

$$C_{i \rightarrow j}^{(s)}(\sigma_i^s | \sigma_j^{s-1}) U^{(s+1)} \Lambda^{(s+1)} \stackrel{\text{SVD}}{=} U^{(s)} \Lambda^{(s)} \tilde{C}_{i \rightarrow j}^{(s)}(\sigma_i^s | \sigma_j^{s-1}), \quad (24)$$

as shown in figure 9 and end at the left boundary with $\tilde{C}_{i \rightarrow j}^{(0)}(\sigma_i^0) := C_{i \rightarrow j}^{(0)}(\sigma_i^0) U^{(1)} \Lambda^{(1)}$ such that all \tilde{C} -tensors except \tilde{C}^0 now obey the right orthonormality constraint. The computation cost for each such SVD is proportional to \tilde{M}^3 .

In a subsequent sweep from left to right, again based on singular value decompositions, we can now do the actual truncations to decrease bond dimensions ($\tilde{C} \rightarrow \bar{C}$), e.g. by setting a threshold ε and truncating all singular values λ_k with $\lambda_k^2 < \varepsilon \cdot \sum_q \lambda_q^2$. All \bar{C} -tensors except $\bar{C}^{(t+1)}$ now obey the left orthonormality constraint and

$$\mu_{i \rightarrow j}(\bar{\sigma}_i^{t+1} | \bar{\sigma}_j^t) = \tilde{C}_{i \rightarrow j}^{(0)}(\sigma_i^0) \left[\prod_{s=1}^{t+1} \tilde{C}_{i \rightarrow j}^{(s)}(\sigma_i^s | \sigma_j^{s-1}) \right] \stackrel{\text{trunc}}{\approx} \bar{C}_{i \rightarrow j}^{(0)}(\sigma_i^0) \left[\prod_{s=1}^{t+1} \bar{C}_{i \rightarrow j}^{(s)}(\sigma_i^s | \sigma_j^{s-1}) \right]. \quad (25)$$

We now need to reorder the indices $\{\sigma_i^s\}$ and $\{\sigma_j^s\}$ of the vertex states. In a sweep from right to left, we go from the index assignment $(\sigma_i^0)(\sigma_i^1 | \sigma_j^0)(\sigma_i^2 | \sigma_j^1) \dots (\sigma_i^{t+1} | \sigma_j^t)$ in the

The matrix product approximation for the dynamic cavity method

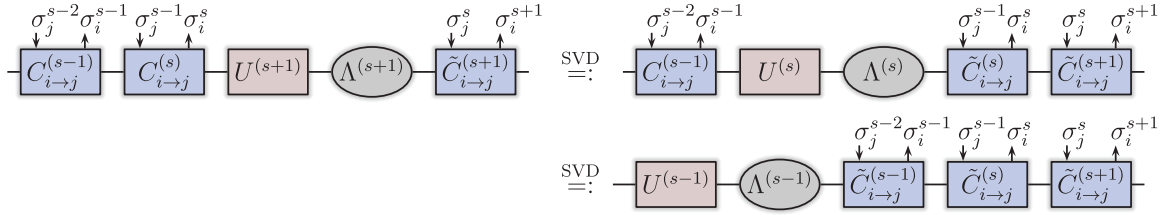


Figure 9. In the truncation scheme described in section 4.2, the evolved MPEM (22) needs to be orthonormalized before nonzero singular values can be truncated in a subsequent sweep. This orthonormalization is accomplished through a sequence of singular value decompositions, sweeping the matrix product from right to left. In the process, the C -tensors (23) are replaced by \tilde{C} -tensors that obey the right orthonormality constraint; see equation (24).

truncated and orthonormalized version of the matrix product in equation (25) to the assignment $(\sigma_i^0 \sigma_j^0)(\sigma_i^1 | \sigma_j^1) \dots (\sigma_i^t | \sigma_j^t)(\sigma_i^{t+1})$ in the matrix product

$$\mu_{i \rightarrow j}(\bar{\sigma}_i^{t+1} | \bar{\sigma}_j^t) \stackrel{\text{trunc}}{\approx} \left[\prod_{s=0}^t D_{i \rightarrow j}^{(s)}(\sigma_i^s | \sigma_j^s) \right] D_{i \rightarrow j}^{(t+1)}(\sigma_i^{t+1}). \quad (26)$$

At the right boundary, we start with a singular value decomposition and controlled truncation (SVT) $\bar{C}_{i \rightarrow j}^{(t+1)}(\sigma_i^{t+1} | \sigma_j^t) \stackrel{\text{SVT}}{\approx} U^{(t+1)}(\sigma_j^t) \Lambda^{(t+1)} D_{i \rightarrow j}^{(t+1)}(\sigma_i^{t+1})$ as described in footnote², continue for the bulk tensors $s = t, t-1, \dots, 1$ with

$$\bar{C}_{i \rightarrow j}^{(s)}(\sigma_i^s | \sigma_j^{s-1}) U^{(s+1)}(\sigma_j^s) \Lambda^{(s+1)} \stackrel{\text{SVT}}{\approx} U^{(s)}(\sigma_j^{s-1}) \Lambda^{(s)} D_{i \rightarrow j}^{(s)}(\sigma_i^s | \sigma_j^s), \quad (27)$$

and end at the left boundary with $D_{i \rightarrow j}^{(0)}(\sigma_i^0 | \sigma_j^0) := \bar{C}_{i \rightarrow j}^{(0)}(\sigma_i^0) U^{(1)}(\sigma_j^0) \Lambda^{(1)}$.

To finally obtain the evolved edge message in canonical form (21), in a sweep from left to right, we go from the index assignment $(\sigma_i^0 \sigma_j^0)(\sigma_i^1 | \sigma_j^1) \dots (\sigma_i^t | \sigma_j^t)(\sigma_i^{t+1})$ in equation (26) to the canonical assignment $(\sigma_j^0)(\sigma_i^0 | \sigma_j^1)(\sigma_i^1 | \sigma_j^2) \dots (\sigma_i^{t-1} | \sigma_j^t)(\sigma_i^t)(\sigma_i^{t+1})$. At the left boundary, we start with $D_{i \rightarrow j}^{(0)}(\sigma_i^0 | \sigma_j^0) \stackrel{\text{SVT}}{\approx} B_{i \rightarrow j}^{(0)}(\sigma_j^0) \Lambda^{(0)} V^{(0)}(\sigma_i^0)$, continue for the bulk tensors $s = 1, 2, \dots, t$ with

$$\Lambda^{(s-1)} V^{(s-1)}(\sigma_i^{s-1}) D_{i \rightarrow j}^{(s)}(\sigma_i^s | \sigma_j^s) \stackrel{\text{SVT}}{\approx} B_{i \rightarrow j}^{(s)}(\sigma_i^{s-1} | \sigma_j^s) \Lambda^{(s)} V^{(s)}(\sigma_i^s), \quad (28)$$

and end at the left boundary with $\Lambda^{(t)} V^{(t)}(\sigma_i^t) D_{i \rightarrow j}^{(t+1)}(\sigma_i^{t+1}) \stackrel{\text{SVT}}{\approx} B_{i \rightarrow j}^{(t+1)}(\sigma_i^t) \Lambda^{(t+1)} V^{(t+1)}(\sigma_i^{t+1})$ and $B_{i \rightarrow j}^{(t+2)}(\sigma_i^{t+1}) := \Lambda^{(t+1)} V^{(t+1)}(\sigma_i^{t+1})$. The matrix products for the different stages in the evolution and truncation of the edge message $\mu_{i \rightarrow j}$ are illustrated in figure 10.

² The resulting matrices \mathcal{U}^\dagger and \mathcal{D} , defined as $\mathcal{U}_{(a,\sigma),k} := [U^{(t+1)}(\sigma)]_{a,k}$ and $\mathcal{D}_{k,(\sigma,b)} := [D_{i \rightarrow j}^{(t+1)}(\sigma)]_{k,b}$ are isometric in the sense that $\mathcal{U}^\dagger \mathcal{U} = \mathbb{1} = \mathcal{D} \mathcal{D}^\dagger$. These matrices are obtained from the singular value decomposition $\bar{C} \approx \mathcal{U} \Lambda \mathcal{D}$ and subsequent truncation of small singular values λ_k , where $\bar{C}_{(a,\sigma),(b,\sigma')} := [\bar{C}_{i \rightarrow j}^{(t+1)}(\sigma | \sigma')]_{a,b}$. The isometric property of \mathcal{D} , corresponds to the right orthonormality constraint as defined in the second part of equation (18b).

The matrix product approximation for the dynamic cavity method

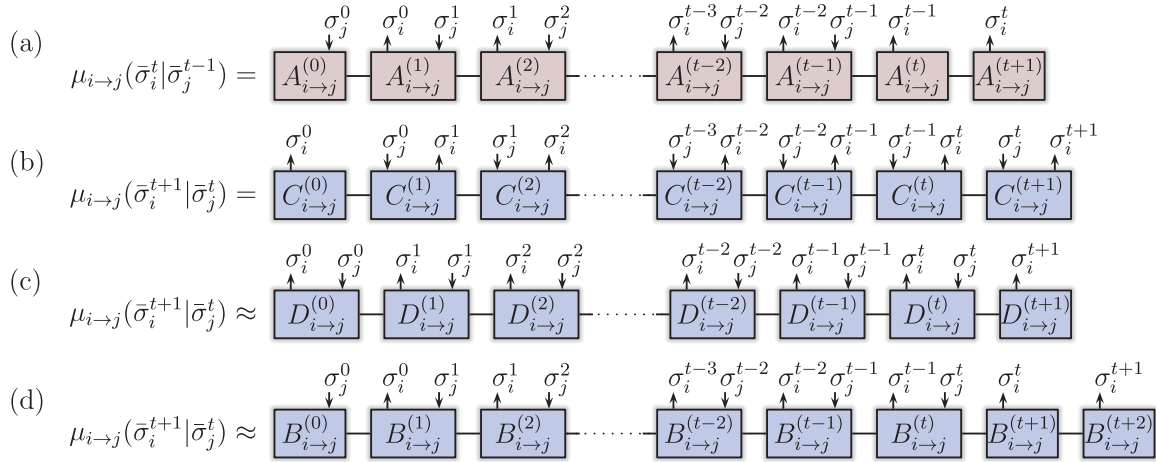


Figure 10. Different forms of MPEMs during a time evolution step $t \rightarrow t + 1$. (a) Canonical form (14) of MPEMs before the evolution step. (b) Non-canonical but exact form (22) of an evolved MPEM. (c) Non-canonical and approximative form (26) of the evolved MPEM after a sequence of truncations and reorderings of vertex variables. (d) Canonical approximative form (21) of the evolved MPEM after another sequence of truncations and vertex variable reorderings.

5. Evaluation of observables

As described in section 2.5, the joint probability of trajectories $\bar{\sigma}_i^t$ and $\bar{\sigma}_j^{t-1}$ for the vertices of an edge (i, j) is given by the product of the two corresponding edge messages, $P(\bar{\sigma}_i^t, \bar{\sigma}_j^t) = \mu_{i \rightarrow j}(\bar{\sigma}_i^t | \bar{\sigma}_j^{t-1}) \mu_{j \rightarrow i}(\bar{\sigma}_j^t | \bar{\sigma}_i^{t-1})$. Given matrix product representations (approximations) of these edge messages in canonical form (14), time-local observables and temporal correlation functions can be evaluated efficiently. In order to evaluate, for example, the probability $P(\sigma_i^t, \sigma_j^t)$ of the edge being in state (σ_i^t, σ_j^t) at time t , we simply contract all indices that occur in both MPEMs as depicted in figure 11. The contractions can be started at the left boundary ($s = 0$) in such a way that the total computation cost scales as $\mathcal{O}(tM^3)$.

To this purpose we start at the left boundary, i.e. at time slice $s = 0$, with the multiplication $E^{(0)}(\sigma_i^0, \sigma_j^0) := A_{i \rightarrow j}^{(0)}(\sigma_j^0) [A_{j \rightarrow i}^{(0)}(\sigma_i^0)]^\top$, continue for the bulk tensors $s = 0, 1, \dots, t - 2$ with

$$E^{(s+1)}(\sigma_i^{s+1}, \sigma_j^{s+1}) := \sum_{\sigma_i^s} A_{i \rightarrow j}^{(s+1)}(\sigma_i^s | \sigma_j^{s+1}) \left(\sum_{\sigma_j^s} E^{(s)}(\sigma_i^s, \sigma_j^s) [A_{j \rightarrow i}^{(s+1)}(\sigma_j^s | \sigma_i^{s+1})]^\top \right), \quad (29)$$

and finish with $E^{(t)} := \sum_{\sigma_i^{t-1}} A_{i \rightarrow j}^{(t)}(\sigma_i^{t-1}) \left(\sum_{\sigma_j^{t-1}} E^{(t-1)}(\sigma_i^{t-1}, \sigma_j^{t-1}) [A_{j \rightarrow i}^{(t)}(\sigma_j^{t-1})]^\top \right)$ and the final step $E^{(t+1)}(\sigma_i^t, \sigma_j^t) := A_{i \rightarrow j}^{(t+1)}(\sigma_i^t) (E^{(t)} [A_{j \rightarrow i}^{(t+1)}(\sigma_j^t)]^\top)$. Now, $E^{(t+1)}(\sigma, \sigma')$ are scalars (1×1 matrices because $M_{t+2} = 1$) that yield the desired joint probability

$$P(\sigma_i^t, \sigma_j^t) = E^{(t+1)}(\sigma_i^t, \sigma_j^t) / \sum_{\sigma, \sigma'} E^{(t+1)}(\sigma, \sigma'). \quad (30)$$

During the time-evolution of the MPEMs (section 4) it is natural to normalize the edge messages according to the 2-norm. In that case, it can be necessary to add a

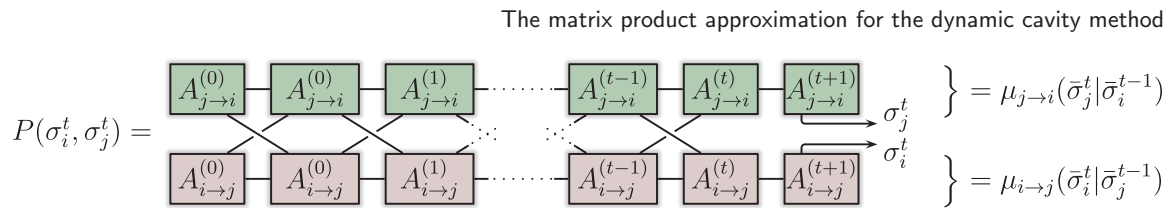


Figure 11. Given MPEMs $\mu_{i \rightarrow j}(\bar{\sigma}_i^t | \bar{\sigma}_j^{t-1})$ and $\mu_{j \rightarrow i}(\bar{\sigma}_j^t | \bar{\sigma}_i^{t-1})$ for the edge (i, j) , observables such as the probability $P(\sigma_i^t, \sigma_j^t)$ of the edge being in state (σ_i^t, σ_j^t) at time t can be computed efficiently by (partial) contraction.

renormalization in equation (29) in order to avoid the generation of very large matrix elements, e.g. by replacing in each step $E^{(s)}$ by $E^{(s)} / \|E^{(s)}\|$.

6. An improved truncation scheme

In order to recast the evolved state (22) into canonical form (21) and truncate bond dimensions, the first step in section 4.2 was to reorthonormalize the tensors ($C \rightarrow \tilde{C}$) in a sweep from right to left before doing truncations in a subsequent sweep from left to right ($\tilde{C} \rightarrow \bar{C}$). The preparatory first sweep can be avoided as described in the following. This also reduces computation costs substantially from $\mathcal{O}(M^{3z-3})$ to $\mathcal{O}(M^{2z-1})$, where M denotes MPEM bond dimensions and z denotes vertex degrees.

In extension of section 3.3, let us first discuss how controlled truncations (of Schmidt coefficients) can be done when the basis $\{|a\rangle\}$ of the left part \mathcal{L} , containing sites $[1, \ell]$, is orthonormal while the basis $\{|b\rangle\}$ for the right part \mathcal{R} , containing sites $[\ell + 1, L]$, is *not* orthonormal. Recall that the purpose of the preparatory first sweep in section 4.2 was to orthonormalize *both* bases. For notational simplicity, let us again use the example of a quantum many-body system with L lattice sites. We are given a state

$$|\psi\rangle =: \sum_{a,b} \psi_{a,b} |a\rangle \otimes |b\rangle \quad (31)$$

with orthonormal states $\langle a|a'\rangle = \delta_{a,a'}$ for the left part and arbitrary states $|b\rangle$ for the right part. Now one can find a good approximation $|\psi_{\text{trunc}}\rangle$ of $|\psi\rangle$ (with respect to the 2-norm distance) by diagonalization of the reduced density matrix $\hat{\varrho}_{\mathcal{L}}$ for the left part. This density matrix,

$$\hat{\varrho}_{\mathcal{L}} \equiv \text{Tr}_{\mathcal{R}} |\psi\rangle \langle \psi| = \sum_{a,a',b,b'} \psi_{a,b} |a\rangle \langle b'| \langle a'| \psi_{a',b'}^*, \quad (32)$$

is obtained by a partial trace over \mathcal{R} . With the overlap matrix $F_{b,b'} := \langle b'|b\rangle$ we have $[\varrho_{\mathcal{L}}]_{a,a'} := \langle a|\hat{\varrho}_{\mathcal{L}}|a'\rangle = [\psi F \psi^\dagger]_{a,a'}$ which can be diagonalized according to

$$\varrho_{\mathcal{L}} = \psi F \psi^\dagger =: U \Lambda^2 U^\dagger, \quad (33)$$

where, as in section 3.3, U is a unitary matrix and Λ is the diagonal matrix containing descendingly ordered Schmidt coefficients $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_{M'}$ (square roots of density matrix eigenvalues).

The matrix product approximation for the dynamic cavity method

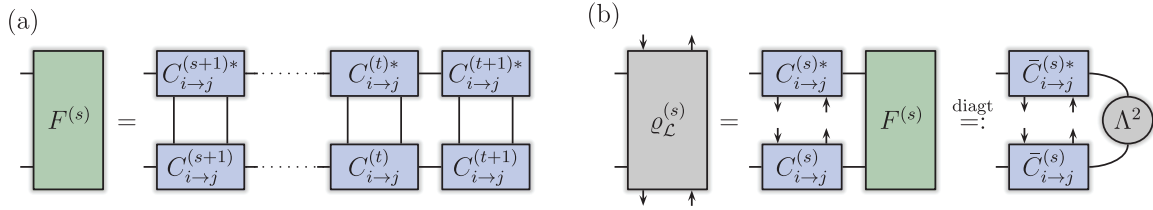


Figure 12. (a) Overlap matrices $F^{(s)}$ are needed in the density matrix truncation scheme discussed in section 6. It allows to truncate without a preparatory orthonormalization sweep and substantially reduces computation costs. Overlap matrices can be computed using the iteration (37). (b) From $F^{(s)}$ and tensor $C^{(s)}$, one obtains the reduced density matrix $\varrho^{(s)}$ (equation (39a)) and its diagonalization and truncation (39b) yields tensor $\tilde{C}^{(s)}$ that obeys the left orthonormality constraint and the singular values $\Lambda^{(s)}$ employed in the truncation.

$$|\psi\rangle = \sum_{a,b,k} U_{a,k} R_{k,b} |a\rangle \otimes |b\rangle = \sum_{k=1}^{M'} |k_{\mathcal{L}}\rangle \otimes |\tilde{k}_{\mathcal{R}}\rangle, \quad (34a)$$

where $\{|k_{\mathcal{L}}\rangle := \sum_a U_{a,k} |a\rangle\}$ is an orthonormal basis for \mathcal{L} and $\{|\tilde{k}_{\mathcal{R}}\rangle := \sum_b R_{k,b} |b\rangle\}$ are orthogonal states with norms λ_k for \mathcal{R} with $R := U^\dagger \psi$.

The approximation after truncating all Schmidt coefficients λ_k with $k > M$ is

$$|\psi_{\text{trunc}}\rangle = \sum_{k=1}^M |k_{\mathcal{L}}\rangle \otimes |\tilde{k}_{\mathcal{R}}\rangle \quad \text{with error} \quad \|\psi - \psi_{\text{trunc}}\|^2 = \sum_{k=M+1}^{M'} \|\tilde{k}_{\mathcal{R}}\|^2 = \sum_{k=M+1}^{M'} \lambda_k^2. \quad (34b)$$

The orthogonality and norms of the states $\{|\tilde{k}_{\mathcal{R}}\rangle\}$ follow as

$$\langle \tilde{k}'_{\mathcal{R}} | \tilde{k}_{\mathcal{R}} \rangle = \sum_{b,b'} R_{k',b'}^* \langle b' | b \rangle R_{k,b} = [R F R^\dagger]_{k,k'} = [U^\dagger \psi F \psi^\dagger U]_{k,k'} = [\Lambda^2]_{k,k'} = \delta_{k,k'} \lambda_k^2. \quad (35)$$

The truncation (34b) is suitable for an MPS $|\psi\rangle$ (and analogously for MPEM) given in the form

$$|\psi\rangle = \sum_{\mathbf{n}} Y_1^{n_1} \cdots Y_\ell^{n_\ell} \cdot A_{\ell+1}^{n_{\ell+1}} \cdots A_L^{n_L} |\mathbf{n}\rangle \quad \text{with} \quad \sum_{\mathbf{n}} (Y_i^n)^\dagger Y_i^n = \mathbb{1}. \quad (36)$$

This is the form (31) with orthonormal basis states $|a\rangle = \sum_{n_1, \dots, n_\ell} [Y_1^{n_1} \cdots Y_\ell^{n_\ell}]_{1,a} |n_1 \dots n_\ell\rangle$ for \mathcal{L} (see equation (20)), non-orthogonal states $|b\rangle = \sum_{n_{\ell+1}, \dots, n_L} [A_{\ell+1}^{n_{\ell+1}} \cdots A_L^{n_L}]_{b,1} |n_{\ell+1} \dots n_L\rangle$ for \mathcal{R} , and $\psi = \mathbb{1}$. The overlap matrix F can be computed in an iteration with $F_L := \mathbb{1}$ and $F_j := \sum_{\mathbf{n}} A_{j+1}^{\mathbf{n}} F_{j+1} (A_{j+1}^{\mathbf{n}})^\dagger$ for $j = L-1, \dots, \ell$. With a diagonalization of $F \equiv F_\ell$, as in equation (33), we then obtain an optimally truncated state $|\psi_{\text{trunc}}\rangle$ in MPS form.

With this type of truncation, the preparatory first orthonormalization sweep for the evolved MPEM (right to left, $C \rightarrow \tilde{C}$), described in section 4.2, can be avoided and computation costs can be reduced substantially. Going from right to left, one can first compute all overlap matrices $F^{(s)}$ for the evolved MPEM (22) with

$$F^{(t+1)} := \mathbb{1} \quad \text{and} \quad F^{(s)} := \sum_{\sigma_i, \sigma_j} C_{i \rightarrow j}^{(s+1)}(\sigma_i | \sigma_j) F^{(s+1)}(C_{i \rightarrow j}^{(s+1)}(\sigma_i | \sigma_j))^\dagger \quad \text{for } s = t, \dots, 0 \quad (37)$$

as illustrated in figure 12(a). With these, we can directly truncate the evolved MPEM, in a sweep from left to right. At the left boundary, we start with

$$\varrho_{\mathcal{L}}^{(0)}(\sigma_i, \sigma'_i) = C_{i \rightarrow j}^{(0)}(\sigma_i) F^{(0)}(C_{i \rightarrow j}^{(0)}(\sigma'_i))^\dagger \stackrel{\text{diagt}}{=} \bar{C}_{i \rightarrow j}^{(0)}(\sigma_i) (\Lambda^{(0)})^2 (\bar{C}_{i \rightarrow j}^{(0)}(\sigma'_i))^\dagger \quad (38a)$$

$$\text{and} \quad X^{(0)} := \sum_{\sigma_i} (\bar{C}_{i \rightarrow j}^{(0)}(\sigma_i))^\dagger C_{i \rightarrow j}^{(0)}(\sigma_i), \quad (38b)$$

where we diagonalize and truncate (diagt) with the diagonalization as in equation (33) and the truncation as in equation (34b). The new tensor $\bar{C}_{i \rightarrow j}^{(0)}$ obeys the left orthonormality constraint (see equation (36)) and we need to include the matrix $X^{(0)}$ right of $\bar{C}_{i \rightarrow j}^{(0)}$ to keep the matrix product invariant such that $C_{i \rightarrow j}^{(0)}(\sigma_i^0) \stackrel{\text{trunc}}{\approx} \bar{C}_{i \rightarrow j}^{(0)}(\sigma_i^0) X^{(0)}$ in equation (22). For the bulk tensors $s = 1, 2, \dots, t$, we continue with

$$\varrho_{\mathcal{L}}^{(s)}(\sigma_i \sigma_j, \sigma'_i \sigma'_j) = X^{(s-1)} C_{i \rightarrow j}^{(s)}(\sigma_i | \sigma_j) F^{(s)}(X^{(s-1)} C_{i \rightarrow j}^{(s)}(\sigma'_i | \sigma'_j))^\dagger \quad (39a)$$

$$\stackrel{\text{diagt}}{=} \bar{C}_{i \rightarrow j}^{(s)}(\sigma_i | \sigma_j) (\Lambda^{(s)})^2 (\bar{C}_{i \rightarrow j}^{(s)}(\sigma'_i | \sigma'_j))^\dagger \quad (39b)$$

$$\text{and} \quad X^{(s)} := \sum_{\sigma_i, \sigma_j} (\bar{C}_{i \rightarrow j}^{(s)}(\sigma_i | \sigma_j))^\dagger C_{i \rightarrow j}^{(s)}(\sigma_i | \sigma_j). \quad (39c)$$

See figure 12(b). We end at the right boundary with $\bar{C}_{i \rightarrow j}^{(t+1)}(\sigma_i | \sigma_j) := X^{(t)} C_{i \rightarrow j}^{(t+1)}(\sigma_i | \sigma_j)$ such that all \bar{C} -tensors except $\bar{C}^{(t+1)}$ now obey the left orthonormality constraint.

Truncating in this way, we arrive at equation (25) of the conceptually simpler truncation scheme of section 4.2 and can continue in the same way as described there, rearranging vertex variables in the evolved MPEM to bring it into the canonical form (21). The major advantage of the more elaborate density matrix truncation scheme is the reduced computation cost. With bond dimensions M of the MPEM (14) before the evolution step, the C -tensors of the exactly evolved MPEM (22) have increased bond dimensions M^{z-1} , where z is the degree of vertex i . The computationally most expensive step in the simple truncation scheme of section 4.2 is the singular value decomposition (24) for the orthonormalization of the C -tensors; the cost scales as $\mathcal{O}(M^{3z-3})$. The most expensive steps in the more efficient density matrix truncation scheme are the computations of overlap matrices (37) and the reduced density matrices (39a). Exploiting the structure of the C -tensors (23), these operations can be accomplished with a cost $\mathcal{O}(M^{2z-1})$ if we assume that the bond dimensions of tensors \bar{C} of the evolved MPEM after truncation are similar to that of the A -tensors in the original MPEM (14), i.e. approximately M . This is generally the case. When we fix a truncation threshold ε for the discarded 2-norm weight as discussed in section 4.2, bond dimensions evolve smoothly in time.

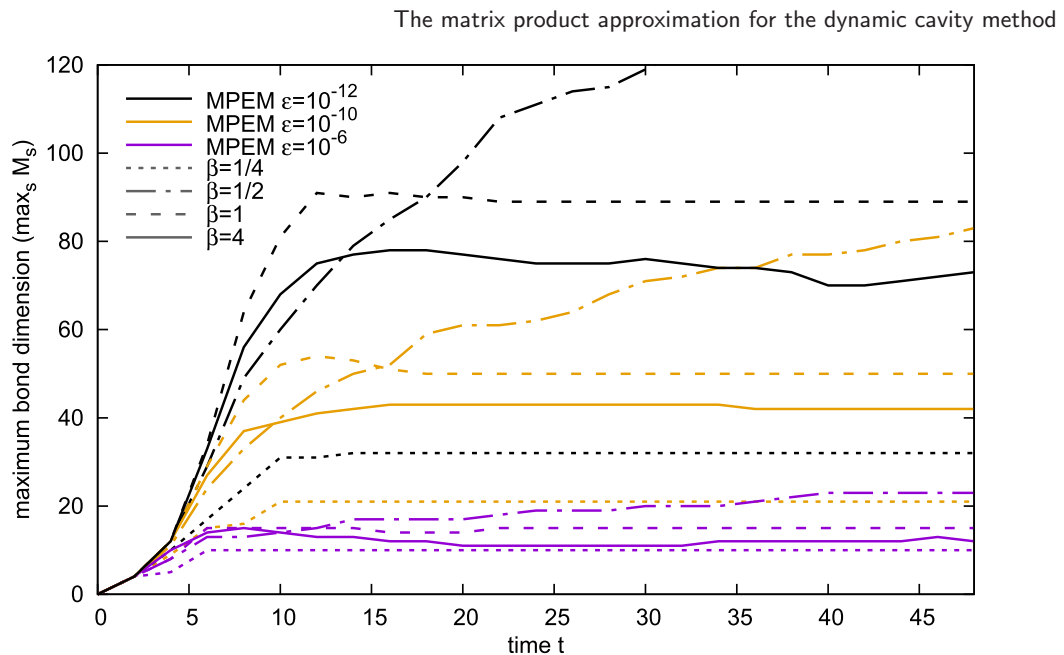


Figure 13. Evolution of maximum MPEM bond dimensions $\max_s M_s$ for Glauber dynamics of the kinetic Ising model (40) on $z=3$ random regular graphs. Truncation thresholds, indicated by color, were fixed to $\varepsilon = 10^{-6}$, 10^{-10} , and 10^{-12} , respectively. Bond dimensions are shown for inverse temperatures $\beta = 4, 1, 1/2$, and $1/4$ (indicated by dash type). Bond dimensions increase with decreasing truncation threshold. They become constant at longer times which implies that computation costs only grow linearly in time.

7. Computation costs in Glauber–Ising dynamics

As the number of possible trajectories on a vertex increases exponentially in time, the computation costs for exact dynamic belief propagation (11) also grow exponentially. As discussed in the introduction, this has, so far, substantially limited the applicability of the dynamic cavity method. The MPEM approach allows for a controlled reduction of the computational complexity, exploiting the decay of temporal correlations to truncate unimportant components of the edge messages.

Concerning computation costs, a decisive question is now, what bond dimensions M_s in MPEMs are required to achieve a certain approximation accuracy in comparison to the exact evolution. Also, how does the required bond dimension evolve with time? Generally, it is to be expected that the maximum bond dimension, $\max_s M_s$, will converge to a constant as a function of time if connected temporal correlations decay exponentially. In simulations, it is often favorable not to fix the truncation dimension, but to fix instead a threshold ε and to truncate all singular values λ_k with $\lambda_k^2 < \varepsilon \cdot \sum_q \lambda_q^2$. The threshold ε controls the 2-norm loss in each truncation. Allowing bond dimensions to evolve accordingly, avoids wasting computation time and gives a measure for the information-theoretic complexity of the edge messages.

As an example, we consider Glauber dynamics of the kinetic Ising model [54] on random regular graphs with vertex degree $z=3$ in the thermodynamic limit. Specifically, Ising spins interact ferromagnetically with local transition matrices

$$w_i(\sigma_i^{t+1}|\sigma_{\partial i}^t) = \frac{1}{Z} \exp\left(\beta \sum_{j \in \partial i} \sigma_i^{t+1} \sigma_j^t\right). \quad (40)$$

At time $t = 0$, all spins have magnetization $\langle \sigma_i^0 \rangle = 1/2$, i.e. $p_i(\uparrow) = 3/4$.

Figure 13 shows the evolution of maximum bond dimensions with time. They increase with decreasing truncation threshold ε . As expected, they become constant at longer times. The number of tensors $A^{(0)}, \dots, A^{(t+1)}$ in an MPEM increases linearly with t , and we need to sweep a few times through the matrix product in each iteration. With converging bond dimensions, this implies that the computation cost per iteration grow only linearly in time, instead of exponentially. For fixed ε , the largest bond dimensions are needed for inverse temperature $\beta = 1/2$. This is so, because it is the β closest to the phase transition in the Ising model. For this temperature, the bond dimensions do actually not yet show saturation on the time interval displayed in figure 13. They will converge at larger t .

8. Models with vertex-state dependence

Except for the algorithms described in sections 4 and 6 our description was generic in the sense that the local transition matrices $w_i = w_i(\sigma_i^{t+1}|\sigma_{\partial i}^t)$ were allowed to depend on the time- t state σ_i^t of vertex i itself, in addition to the states on neighboring vertices. However, in sections 4 and 6 we considered the case where ∂i does not contain i itself, i.e. that w_i is independent of σ_i^t . Here, we generalize to MPEM algorithms that allow for the dependence on σ_i^t . This is important for many applications like stochastic models for the dynamics of infectious diseases [14, 55, 56] or of opinions in social networks [57]. Naturally, transition matrices with dependence on σ_i^t also arise in time-discretized versions of stochastic continuous-time dynamics, i.e. all models that have a well-defined continuum-time limit as, for example, the Glauber dynamics of Ising spin systems [54]. In particular, when decreasing the time step Δt , one should have

$$w_i(\sigma_i^{t+1}|\sigma_{\partial i}^t) \rightarrow \delta_{\sigma_i^{t+1}, \sigma_i^t} \quad \text{for } \Delta t \rightarrow 0. \quad (41)$$

Vertex i is now contained in its neighborhood $\partial i = \{i, j, k_1, \dots, k_{z-1}\}$, where z denotes the vertex degree. In this more general scenario, the evolved MPEM (22) cannot be constructed in an entirely local fashion anymore. Instead, the \tilde{C} -tensors are constructed in a sweep from right ($s = t + 1$) to left ($s = 0$), imposing at the same time the right orthonormality constraint, $\sum_{\sigma, \sigma'} \tilde{C}_{i \rightarrow j}^{(s)}(\sigma|\sigma') [\tilde{C}_{i \rightarrow j}^{(s)}(\sigma|\sigma')]^\dagger = \mathbb{1}$ (see equation (18b)).

We start at the right boundary ($s = t + 1$) by doing a singular value decomposition (SVD) of the tensor

$$C_{i \rightarrow j}^{(t+1)}(\sigma_i^{t+1}|\sigma_i^t, \sigma_j^t) := \sum_{\sigma_{k_1}^t \dots \sigma_{k_{z-1}}^t} w(\sigma_i^{t+1}|\sigma_{\partial i}^t) \left[\bigotimes_{k=1}^{z-1} A_{k_n \rightarrow i}^{(t+1)}(\sigma_{k_n}^t) \right] \stackrel{\text{SVD}}{=} U^{(t+1)}(\sigma_i^t) \Lambda^{(t+1)} \tilde{C}_{i \rightarrow j}^{(t+1)}(\sigma_i^{t+1}|\sigma_j^t)$$

in order to obtain the tensor $\tilde{C}_{i \rightarrow j}^{(t+1)}(\sigma_i^{t+1}|\sigma_j^t)$, where $\Lambda^{(t+1)}$ is the diagonal matrix of singular values, $\tilde{C}_{i \rightarrow j}^{(t+1)}$ obeys the right orthonormality constraint, and, similarly, $\sum_{\sigma} [U^{(t+1)}(\sigma)]^\dagger U^{(t+1)}(\sigma) = \mathbb{1}$. For time $s = t$, the process continues with

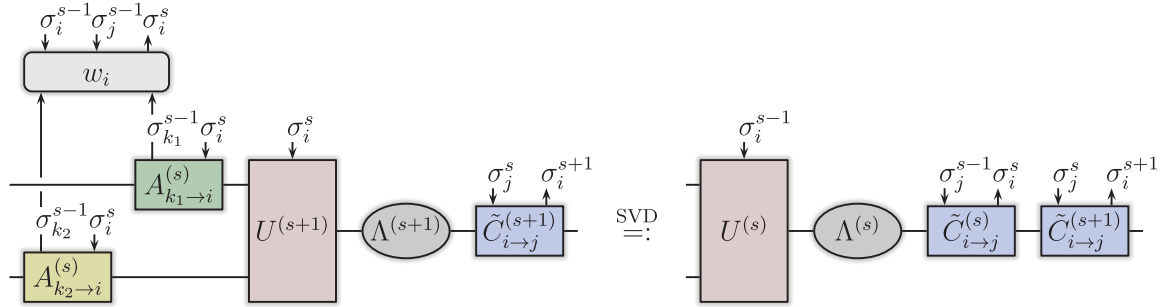


Figure 14. In section 8, we discuss the case where the transition matrix $w_i = w_i(\sigma_i^{t+1} | \sigma_{\partial i}^t)$ for vertex i is allowed to also depend on the time- t state σ_i^t of vertex i itself, instead of just depending on states of nearest neighbors. In this case, C -tensors of the evolved MPEM (22) cannot be constructed in an entirely local fashion anymore. Instead, the \tilde{C} -tensors that also obey the right orthonormality constraint are constructed in a sweep from right ($s = t + 1$) to left ($s = 0$), sequentially doing singular value decompositions and contracting with A -tensors from MPEMs of the previous time step. The iteration, corresponding to equation (43), is shown for a vertex with degree $z = 3$. Compare to figure 9 for the case without vertex-state dependence.

$$C_{i \rightarrow j}^{(t)}(\sigma_i^t | \sigma_i^{t-1}, \sigma_j^{t-1}) := \sum_{\sigma_{k_1}^{t-1} \dots \sigma_{k_{z-1}}^{t-1}} w(\sigma_i^t | \sigma_{\partial i}^{t-1}) \left[\bigotimes_{n=1}^{z-1} A_{k_n \rightarrow i}^{(t)}(\sigma_{k_n}^{t-1}) \right] U^{(t+1)}(\sigma_i^t) \Lambda^{(t+1)} \\ \stackrel{\text{SVD}}{=} U^{(t)}(\sigma_i^{t-1}) \Lambda^{(t)} \tilde{C}_{i \rightarrow j}^{(t)}(\sigma_i^t | \sigma_j^{t-1}). \quad (42)$$

For the bulk tensors, $s = t, t - 1, \dots, 1$, the corresponding equations read very similarly

$$C_{i \rightarrow j}^{(s)}(\sigma_i^s | \sigma_i^{s-1}, \sigma_j^{s-1}) := \sum_{\sigma_{k_1}^{s-1} \dots \sigma_{k_{z-1}}^{s-1}} w(\sigma_i^s | \sigma_{\partial i}^{s-1}) \left[\bigotimes_{n=1}^{z-1} A_{k_n \rightarrow i}^{(s)}(\sigma_{k_n}^{s-1} | \sigma_i^s) \right] U^{(s+1)}(\sigma_i^s) \Lambda^{(s+1)} \\ \stackrel{\text{SVD}}{=} U^{(s)}(\sigma_i^{s-1}) \Lambda^{(s)} \tilde{C}_{i \rightarrow j}^{(s)}(\sigma_i^s | \sigma_j^{s-1}) \quad (43)$$

as illustrated in figure 14. The process ends at the left boundary with the assignment

$$\tilde{C}_{i \rightarrow j}^{(0)}(\sigma_i^0) := p_i(\sigma_i^0) \left[\bigotimes_{n=1}^{z-1} A_{k_n \rightarrow i}^{(0)}(\sigma_{k_n}^0) \right] U^{(1)}(\sigma_i^0) \Lambda^{(1)}. \quad (44)$$

The remaining truncation and vertex variable reordering sweeps, done in order to transform the non-canonical form (22) into the canonical form (21), can be executed exactly as discussed in the somewhat simpler situation covered in section 4.2.

The more efficient density matrix truncation scheme discussed in section 6, similarly, can be adapted to the situation where transition matrix $w_i = w_i(\sigma_i^{t+1} | \sigma_{\partial i}^t)$ depends on σ_i^t . Like the unitaries $U^{(s)}$ in equations (42)–(44), the $X^{(s)}$ -matrices in equations (38) and (39) of the density matrix truncation scheme will then depend on σ_i^{s-1} .

Continuous-time stochastic dynamics on locally tree-like networks can be simulated with the described approach after discretization of the time axis using a small time step Δt . It should also be possible to, alternatively, work with continuous matrix products

[58, 59] as introduced in the context of condensed matter physics. This however, entails some technical difficulties and one can expect that using discrete-time MPEMs and a small time step should be the best option. Similarly, quantum many-body systems in continuous real-space have been treated efficiently using MPS with a sufficiently fine space discretization [60, 61]. Other approximative schemes to simulate continuous-time stochastic dynamics are the dynamical replica analysis [62] that captures macroscopic observables and the cavity master equation method [63] that operates on local conditional probabilities.

9. Discussion

The described MPEM algorithm for the simulation of stochastic dynamics is based on the cavity method, applicable for locally tree-like interaction graphs, and on the matrix product approximation for edge messages which exploits the decay of (connected) temporal correlations. The MPEM method lifts restrictions of earlier approaches for the solution of dynamic cavity equations, mentioned in the introduction. It can also be used to simulate in the thermodynamic limit. Compared to Monte Carlo simulations, errors decrease much faster as a function of the invested computation time. This has been demonstrated for Glauber dynamics of the kinetic Ising model in [39]. Hence, the MPEM method is particularly suited for the precise analysis of temporal correlations, decay processes, and low-probability events. For the Glauber–Ising dynamics and fixed truncation thresholds, we have seen here that MPEM bond dimensions converge to a constant as a function of time. This should generally apply when connected temporal correlations decay exponentially. It implies that required computation costs per iteration grow only linearly instead of exponentially in time.

There are several ways in which the MPEM method can be developed further. Here, we have discussed a more efficient truncation scheme that reduces computation costs considerably from $\mathcal{O}(M^{3z-3})$ to $\mathcal{O}(M^{2z-1})$, and we have generalized the method to models where the transition probabilities for a vertex depend explicitly on the vertex state at the previous time step, in addition to the states for its nearest neighbors.

Efficient codes for the MPEM algorithms, including the optimized truncation scheme of section 6, are available from the author and under www.manyparticle.org/~barthel/mpem. They were, for example, employed for the simulations in [39]. We are also happy to collaborate on specific applications.

Acknowledgments

I gratefully acknowledge discussions with Silvio Franz and Caterina De Bacco that initiated this work and support through US Department of Energy grant DE-SC0019449.

References

- [1] Van Kampen N G 2007 *Stochastic Processes in Physics and Chemistry (North-Holland Personal Library)* 3rd edn (Amsterdam: Elsevier)

- [2] Freund J A and Pöschel T 2010 *Stochastic Processes in Physics, Chemistry, and Biology* 9 *Lecture Notes in Physics* (Heidelberg: Springer)
- [3] Capasso V and Bakstein D 2004 *An Introduction to Continuous Time Stochastic Processes: Theory, Models, and Applications to Finance, Biology, and Medicine* (Boston, MA: Birkhäuser)
- [4] Aït-Sahalia Y and Jacod J 2014 *High-Frequency Financial Econometrics* (Princeton, NJ: Princeton University Press)
- [5] Barrat A, Barthélemy M and Vespignani A 2008 *Dynamical Processes on Complex Networks* (New York: Cambridge University Press)
- [6] Newman M E J 2010 *Networks: an Introduction* (Oxford: Oxford University Press)
- [7] Mézard M and Montanari A 2009 *Information, Physics, and Computation* (New York: Oxford University Press)
- [8] Mézard M, Parisi G and Virasoro A 1986 SK model: the replica solution without replicas *Europhys. Lett.* **1** 77
- [9] Mézard M and Parisi G 2001 The Bethe lattice spin glass revisited *Eur. Phys. J. B* **20** 217
- [10] Mézard M, Parisi G and Zecchina R 2002 Analytic and algorithmic solution of random satisfiability problems *Science* **297** 812
- [11] Mézard M and Zecchina R 2002 Random K -satisfiability problem: from an analytic solution to an efficient algorithm *Phys. Rev. E* **66** 056126
- [12] Rogers T, Castillo I P, Kühn R and Takeda K 2008 Cavity approach to the spectral density of sparse symmetric random matrices *Phys. Rev. E* **78** 031116
- [13] Neri I and Bollé D 2009 The cavity approach to parallel dynamics of Ising spins on a graph *J. Stat. Mech.* **P08009**
- [14] Karrer B and Newman M E J 2010 Message passing approach for general epidemic models *Phys. Rev. E* **82** 016101
- [15] Kanoria Y and Montanari A 2011 Majority dynamics on trees and the dynamic cavity method *Ann. Appl. Probab.* **21** 1694
- [16] Shrestha M, Scarpino S V and Moore C 2015 Message-passing approach for recurrent-state epidemic models on networks *Phys. Rev. E* **92** 022821
- [17] Castellano C and Pastor-Satorras R 2018 Relevance of backtracking paths in recurrent-state epidemic spreading on network *Phys. Rev. E* **98** 052313
- [18] Lokhov A Y, Mézard M and Zdeborová L 2015 Dynamic message-passing equations for models with unidirectional dynamics *Phys. Rev. E* **91** 012811
- [19] Lokhov A Y, Mézard M, Ohta H and Zdeborová L 2014 Inferring the origin of an epidemic with a dynamic message-passing algorithm *Phys. Rev. E* **90** 012801
- [20] Altarelli F, Braunstein A, Dall'Asta L, Lage-Castellanos A and Zecchina R 2014 Bayesian inference of epidemics on networks via belief propagation *Phys. Rev. Lett.* **112** 118701
- [21] Altarelli F, Braunstein A, Dall'Asta L, Ingrosso A and Zecchina R 2014 The patient-zero problem with noisy observations *J. Stat. Mech.* **P10016**
- [22] Shrestha M and Moore C 2014 Message-passing approach for threshold models of behavior in networks *Phys. Rev. E* **89** 022805
- [23] Lokhov A Y and Saad D 2017 Optimal deployment of resources for maximizing impact in spreading processes *Proc. Natl Acad. Sci. USA* **114** E8138
- [24] Aurell E and Mahmoudi H 2011 A message-passing scheme for non-equilibrium stationary states *J. Stat. Mech.* **P04014**
- [25] Aurell E and Mahmoudi H 2012 Dynamic mean-field and cavity methods for diluted Ising systems *Phys. Rev. E* **85** 031119
- [26] Zhang P 2012 Inference of kinetic Ising model on sparse graphs *J. Stat. Phys.* **148** 502
- [27] Del Ferraro G and Aurell E 2015 Dynamic message-passing approach for kinetic spin models with reversible dynamics *Phys. Rev. E* **92** 010102
- [28] Pelizzola A 2013 Variational approximations for stationary states of Ising-like models *Eur. Phys. J. B* **86** 120
- [29] Vázquez E D, Ferraro G D and Ricci-Tersenghi F 2017 A simple analytical description of the non-stationary dynamics in Ising spin systems *J. Stat. Mech.* **033303**
- [30] Pelizzola A and Pretti M 2017 Variational approximations for stochastic dynamics on graphs *J. Stat. Mech.* **073406**
- [31] Roudi Y and Hertz J 2011 Dynamical TAP equations for non-equilibrium Ising spin glasses *J. Stat. Mech.* **P03031**
- [32] Bachschmid-Romano L, Battistin C, Oppen M and Roudi Y 2016 Variational perturbation and extended Plefka approaches to dynamics on random networks: the case of the kinetic Ising model *J. Phys. A: Math. Theor.* **49** 434003

- [33] Hatchett J P L, Wemmenhove B, Castillo I P, Nikolettopoulos T, Skantzos N S and Coolen A C C 2004 Parallel dynamics of disordered Ising spin systems on finitely connected random graphs *J. Phys. A: Math. Gen.* **37** 6201
- [34] Mimura K and Coolen A C C 2009 Parallel dynamics of disordered Ising spin systems on finitely connected directed random graphs with arbitrary degree distributions *J. Phys. A: Math. Theor.* **42** 415001
- [35] Mozeika A and Saad D 2011 Dynamics of Boolean networks: an exact solution *Phys. Rev. Lett.* **106** 214101
- [36] Mozeika A and Saad D 2012 Phase transitions and memory effects in the dynamics of Boolean networks *Phil. Mag.* **92** 210
- [37] Coolen A C C and Takeda K 2012 Transfer operator analysis of the parallel dynamics of disordered Ising chains *Phil. Mag.* **92** 64
- [38] Mahmoudi H and Saad D 2014 Generalized mean field approximation for parallel dynamics of the Ising model *J. Stat. Mech.* **P07001**
- [39] Barthel T, De Bacco C and Franz S 2018 A matrix product algorithm for stochastic dynamics on networks, applied to non-equilibrium Glauber dynamics *Phys. Rev. E* **97** 010104
- [40] Accardi L 1981 Topics in quantum probability *Phys. Rep.* **77** 169
- [41] Fannes M, Nachtergaele B and Werner R F 1992 Finitely correlated states on quantum spin chains *Commun. Math. Phys.* **144** 443
- [42] Rommer S and Östlund S 1997 A class of ansatz wave functions for 1D spin systems and their relation to DMRG *Phys. Rev. B* **55** 2164
- [43] Schollwöck U 2011 The density-matrix renormalization group in the age of matrix product states *Ann. Phys.* **326** 96
- [44] White S R 1992 Density-matrix formulation for quantum renormalization-groups *Phys. Rev. Lett.* **69** 2863
- [45] White S R 1993 Density-matrix algorithms for quantum renormalization-groups *Phys. Rev. B* **48** 10345
- [46] Schollwöck U 2005 Density-matrix renormalization-group *Rev. Mod. Phys.* **77** 259
- [47] Hastings M B 2007 Entropy, entanglement in quantum ground states *Phys. Rev. B* **76** 035114
- [48] Brandão F G S L and Horodecki M 2013 An area law for entanglement from exponential decay of correlations *Nat. Phys.* **9** 721
- [49] Verstraete F and Cirac J I 2006 Matrix product states represent ground states faithfully *Phys. Rev. B* **73** 094423
- [50] Barthel T 2017 Typical one-dimensional quantum systems at finite temperatures can be simulated efficiently on classical computers (arXiv:1708.09349)
- [51] Oseledets I V 2011 Tensor-train decomposition *SIAM J. Sci. Comput.* **33** 2295
- [52] Golub G H and Van Loan C F 1996 *Matrix Computations* 3rd edn (Baltimore, MD: Johns Hopkins University Press)
- [53] Stewart G 1993 On the early history of the singular value decomposition *SIAM Rev.* **35** 551
- [54] Glauber R J 1963 Time-dependent statistics of the Ising model *J. Math. Phys.* **4** 294
- [55] Murray J D 1989 *Mathematical Biology* (New York: Springer)
- [56] Dangerfield C E, Ross J V and Keeling M J 2009 Integrating stochasticity and network structure into an epidemic model *J. R. Soc. Interface* **6** 761
- [57] Castellano C, Fortunato S and Loreto V 2009 Statistical physics of social dynamics *Rev. Mod. Phys.* **81** 591
- [58] Verstraete F and Cirac J I 2010 Continuous matrix product states for quantum fields *Phys. Rev. Lett.* **104** 190405
- [59] Osborne T J, Eisert J and Verstraete F 2010 Holographic quantum states *Phys. Rev. Lett.* **105** 260401
- [60] Stoudenmire E M, Wagner L O, White S R and Burke K 2012 One-dimensional continuum electronic structure with the density-matrix renormalization group and its implications for density-functional theory *Phys. Rev. Lett.* **109** 056402
- [61] Dolfi M, Bauer B, Troyer M and Ristivojevic Z 2012 Multigrid algorithms for tensor network states *Phys. Rev. Lett.* **109** 020604
- [62] Mozeika A and Coolen A C C 2008 Dynamical replica analysis of processes on finitely connected random graphs: I. Vertex covering *J. Phys. A: Math. Theor.* **41** 115003
- [63] Aurell E, Del Ferraro G, Domínguez E and Mulet R 2017 Cavity master equation for the continuous time dynamics of discrete-spin models and *Phys. Rev. E* **95** 052119