

Stochastic Series Expansion Quantum Monte Carlo

Roger G. Melko

¹ Department of Physics and Astronomy, University of Waterloo, Ontario, N2L 3G1, Canada rgmelko@uwaterloo.ca

² Perimeter Institute for Theoretical Physics, Waterloo, Ontario N2L 2Y5, Canada

Abstract. This Chapter outlines the fundamental construction of the Stochastic Series Expansion, a highly efficient and easily implementable quantum Monte Carlo method for quantum lattice models. Originally devised as a finite-temperature simulation based on a Taylor expansion of the partition function, the method has recently been recast in the formalism of a zero-temperature projector method, where a large power of the Hamiltonian is applied to a trial wavefunction to project out the ground-state. Although these two methods appear formally quite different, their implementation via non-local *loop* or *cluster* algorithms reveals their underlying fundamental similarity. Here, we briefly review the finite- and zero-temperature formalisms, and discuss concrete manifestations of the algorithm for the spin 1/2 Heisenberg and transverse field Ising models.

1.1 Introduction

In the quest to understand the phenomenon of quantum systems in the thermodynamic limit, it is often one's ambition to simulate microscopic models on lattices of the largest sizes practically possible. In the context of quantum Monte Carlo (QMC), this means constructing a simulation methodology that is highly efficient, with favorable scaling properties, free of systematic errors or bias, yet capable of attacking the cornucopia of interesting quantum models which entice modern condensed matter, materials, atomic, molecular and optics, and quantum information scientists. Since QMC involves mapping a D -dimensional quantum system to $D + 1$ dimensions, the symmetry of the quantum Hamiltonian is often encoded directly into the structure of the simulation cell. Therefore, in order to facilitate a practitioner's ability to examine a variety of physical Hamiltonians of interest, a QMC program should also be transparent and easy to implement, allowing flexibility to code in a wide range of different lattice models.

For many applications, the industry standard for simulation of lattice models of quantum spins and bosons (without the sign problem) has become SSE

QMC, which stands for Stochastic Series Expansion.³ Often, SSE simulations for ‘typical’ Hamiltonians can be straightforwardly implemented in a few thousand lines of code. This allows the programmer a high degree of control, facilitates optimization, and encourages algorithmic development. SSE programs scale with high efficiency, typically linear in the number of lattice sites and the inverse temperature: $\mathcal{O}(N\beta)$. This combination of favorable quantities enables a power, utility, and ease of use that is testified to by SSE QMC’s widespread adoption in a variety of applications in the physical sciences.

Modern SSE is based on some of the same principles as Handscomb’s method, namely Taylor expanding the partition function [1]. However, Handscomb (and others after him) only considered cases where the trace of the Hamiltonian could be computed analytically – a fact that caused the method to be considered limited in scope. The major advance, which propelled Handscomb’s method towards wide-spread applicability, was the realization that this trace could be stochastically *sampled*, like in worldline methods. Thus dubbed SSE, Sandvik [2, 3] pioneered not only the original, spin 1/2 version (used extensively to study the Heisenberg model [4]), but many extensions of the versatile framework, including the important recent recasting as a $T = 0$ projector method [5, 6]. The SSE method has been applied to both spin 1/2 and higher spin models [2, 7] on a large variety of lattices, including partially-frustrated systems [8], models with long-range interactions [9], and a whole host of other manifestations of quantum magnetism. It has been adapted to boson Hamiltonians, both hard-core and soft-core, which has caused a proliferation of application on models of cold atoms trapped in optical lattices [10–12]. It has been generalized to study $SU(N)$ symmetric models [13, 14], which has been instrumental in making connection with field-theoretic studies of universality and quantum phase transitions [15, 16]. Today, SSE simulations can routinely access simulation cells with a space-time volume $N\beta$ larger than 10^7 [17]; its limitations it seems are only the imagination of the practitioner.⁴

The above are but a few examples of the success of SSE over the past twenty years. An exhaustive list would quickly fill all of the pages available in this volume. Instead, in this Chapter, we describe the current broad framework of understanding of the SSE QMC method, outlining the modern approach to designing and implementing both $T > 0$ and $T = 0$ programs. The two simplest cases for spin 1/2 models, the isotropic Heisenberg model, and the Ising model in an applied transverse field, are discussed in detail.

1.2 Quantum Monte Carlo formalism

We begin by reviewing the foundational basis of the SSE methodology. Like any Monte Carlo method, the goal of SSE is to construct an importance

³ or alternatively, *Sandvik’s Simple and Efficient QMC*

⁴ and, of course, the infamous “sign problem”

sampling scheme, which ultimately leads to the computation of expectation values in the form of an arithmetic mean,

$$\langle \mathcal{O} \rangle = \frac{1}{N_{\text{mc}}} \sum_{t=0}^{N_{\text{mc}}} \mathcal{O}(x^t). \quad (1.1)$$

Here x^t is the configuration of the simulation cell at ‘time’ step t in a Markov Chain of length $N_{\text{mc}} \rightarrow \infty$. Each time step is one element of a random walk in a higher-dimensional configuration space, weighted with a probability distribution $W(x)$ (where each $W(x^t) > 0$), which samples all possible configurations when the number of steps is infinite. Because the random walk is an importance sampling through this weighted space, the arithmetic mean for finite- N_{mc} corresponds to the estimate,

$$\langle \mathcal{O} \rangle = \frac{\sum_x \mathcal{O}_x W(x)}{\sum_x W(x)}. \quad (1.2)$$

This expectation value is the basis of many Markov Chain Monte Carlo (MCMC) procedures [18]. For the purposes of this Chapter, the form of this expectation value provides a unified starting point for two different QMC formalisms with separate goals: calculating physical estimators at finite temperature, or calculating physical estimators at zero temperature.

We note here that the procedure we call “quantum Monte Carlo” simulation actually consists of two distinct steps [19]. First, we must derive the appropriate *representation* of the quantum mechanical model (i.e. lattice, basis states, and operators) such that this representation can be coded appropriately on a classical computer. This involves mapping a D -dimensional quantum lattice model to a $D + 1$ -dimensional classical representation, which can be thought of as a highly non-trivial classical statistical mechanics problem. Strictly speaking, SSE can be thought of as a representation of $\exp(-\beta H)$ together with some lattice basis; other representations exist, such as continuous-time or interaction representations [19]. Second, we must devise *updating schemes* that sample configurations of this representation, usually through some Metropolis procedure. Similar updating schemes may be used for different representations. In the next two sections, we will discuss the finite- T and $T = 0$ representations. In Section 1.2.3, we’ll outline some basic ideas behind updating schemes employed in the SSE, emphasizing the role of non-local updates.

1.2.1 Finite-temperature representation

For finite-temperature, Equation 1.2 corresponds to the thermal average,

$$\langle \mathcal{O} \rangle = \frac{1}{Z} \text{Tr}\{\mathcal{O} e^{-\beta H}\}, \quad (1.3)$$

where $\beta = 1/T$, and the denominator is the partition function,

$$Z = \sum_x W(x) = \text{Tr}\{e^{-\beta H}\}. \quad (1.4)$$

The first step in the SSE approach is to write the trace in the partition function as a sum over diagonal matrix elements in a basis $\{\alpha_0\}$. Then, one can Taylor expand the exponential to get the expression,

$$Z = \text{Tr}\{e^{-\beta H}\} = \sum_{\alpha_0} \left\langle \alpha_0 \left| \sum_{n=0}^{\infty} \frac{\beta^n}{n!} (-H)^n \right| \alpha_0 \right\rangle. \quad (1.5)$$

We next insert a set of complete basis states as resolutions of the identity, $\sum_{\alpha} |\alpha\rangle\langle\alpha|$ between the n products of $-H$;

$$Z = \sum_{\{\alpha_i\}} \sum_{n=0}^{\infty} \frac{\beta^n}{n!} \langle \alpha_0 | -H | \alpha_1 \rangle \langle \alpha_1 | -H | \alpha_2 \rangle \cdots \langle \alpha_{n-1} | -H | \alpha_n \rangle, \quad (1.6)$$

where, importantly, $\alpha_n = \alpha_0$ to keep the trace nonzero. Note the set of basis states $\{\alpha_i\}$ in the sum is practically impossible to evaluate exactly; this motivates the use of an importance-sampling scheme. Then, the “weight” of a configuration, $W(x)$, is derived from this expression, and can be seen to be proportional to the product of n matrix elements. Each individual matrix element $\langle \alpha_i | -H | \alpha_{i+1} \rangle$ is evaluated as a real number, and must be *positive* to be interpreted as a probability for use in a Metropolis scheme.

Note however that it is very possible for the above matrix elements to be negative, depending on the precise form of H . This is a manifestation of the so called “sign problem” [20]. It can essentially be avoided only in the case where all terms in the Hamiltonian have (or can be made to have) a negative sign in front of them. This can happen either intrinsically, e.g. with ferromagnetic interactions, or through a clever basis rotation, which is possible e.g. for antiferromagnetic interactions on bipartite lattices. In the examples we will consider in this Chapter, resolution of the sign problem will be straightforward – however in general quantum lattice models, it can be quite inhibitive.

The next step is to write the Hamiltonian as a sum of elementary lattice operators,

$$H = - \sum_t \sum_a H_{t,a}, \quad (1.7)$$

where in our chosen representation the operators propagate the basis $H_{t,a} |\alpha_i\rangle \rightarrow |\alpha_{i+1}\rangle$, and all of the $H_{t,a}$ are positive. The indices t and a refer to the operator “types” and the “lattice units” (e.g. bonds) over which the terms will be sampled (specific examples will be given below). We write the partition function as,

$$Z = \sum_{\{\alpha_i\}} \sum_{n=0}^{\infty} \sum_{S_n} \frac{\beta^n}{n!} \prod_{i=1}^n \langle \alpha_{i-1} | H_{t_i, a_i} | \alpha_i \rangle, \quad (1.8)$$

where S_n denotes a sequence of operator-indices specifying the n operators,

$$S_n = [t_1, a_1], [t_2, a_2], \dots, [t_n, a_n], \quad (1.9)$$

which must be importance sampled, along with the basis states α , and the expansion power n .

As the final step, one can truncate the Taylor expansion at a maximum power M . This is not strictly necessary in principle, however it significantly facilitates implementation of the updating scheme. The truncation can be justified by ensuring that the chosen M is always greater than the largest n to occur in a numerical simulation, $M > n_{\max}$, for a given parameter set [4]. Working with a fixed M is possible to do if one inserts $M - n$ “fill-in”, or null, operators $H_{0,0} \equiv I$ into the operator list (where I is the identity operator). However, these operators do not occur in the evaluation of the partition function, and must be accounted for by dividing the final expression for Z by their contribution. To do so note that, statistically, the number of different way of picking the placement of the null operators in the expansion list is given by the binomial coefficient, $\binom{M}{n} = M!/(M-n)!n!$. One is required to divide our truncated partition function by this value, giving,

$$Z = \sum_{\alpha} \sum_{S_M} \frac{(\beta)^n (M-n)!}{M!} \prod_{i=1}^M \langle \alpha_{i-1} | H_{t_i, a_i} | \alpha_i \rangle, \quad (1.10)$$

where the sum over n is now implicitly included in the sampling of S_M .

We thus arrive at the final expression for the representation of the finite-temperature partition function in SSE, which can formally be related to path integral or worldline representations – resulting in the propagation (or expansion) dimension being identified with the imaginary time direction [19,21] With this SSE representation, we are free to devise updating methods adopted from classical Monte Carlo routines to generate the appropriate Markov Chain in the unified space of basis state (α) and operator/worldline (S_M) configurations. This will be discussed in Section 1.2.3; first, however, we introduce the formally different $T = 0$ starting point for QMC that, nonetheless, results in a very similar SSE-type representation, amenable to similar updating techniques as the $T > 0$ representation here.

1.2.2 Zero-temperature projector representation

At $T = 0$, Equation 1.2 can alternatively be used as an estimate for the operator expectation value,

$$\langle \mathcal{O} \rangle = \frac{1}{Z} \langle \Psi | \mathcal{O} | \Psi \rangle, \quad (1.11)$$

where one aims to use some procedure to find Ψ as the ground-state wavefunction of a Hamiltonian [5]. The denominator of Equation 1.2 is then the normalization, or inner product,

$$Z = \langle \Psi | \Psi \rangle. \quad (1.12)$$

In a “projector” QMC representation, the ground state wavefunction is estimated by a procedure where a large power⁵ of the Hamiltonian is applied to a *trial* state, call it $|\alpha\rangle$. This can be seen by writing the trial state in terms of energy eigenstates $|n\rangle$, $n = 0, 1, 2, \dots$, $|\alpha\rangle = \sum_n c_n |n\rangle$, so that a large power of the Hamiltonian will project out the groundstate,

$$\begin{aligned} (-H)^m |\alpha\rangle &= c_0 |E_0|^m \left[|0\rangle + \frac{c_1}{c_0} \left(\frac{E_1}{E_0} \right)^m |1\rangle \dots \right], \\ &\rightarrow c_0 |E_0|^m |0\rangle \text{ as } m \rightarrow \infty. \end{aligned} \quad (1.13)$$

Here, we have assumed that the magnitude of the lowest eigenvalue $|E_0|$ is *largest* of all the eigenvalues. To achieve this, one may be forced to add a sufficiently large negative constant to the overall Hamiltonian (that we have not explicitly included). Then, from this expression, one can write the normalization of the groundstate wavefunction, $Z = \langle 0|0\rangle$ with two projected states (bra and ket) as,

$$Z = \langle \alpha | (-H)^m (-H)^m | \alpha \rangle, \quad (1.14)$$

for large m . The Hamiltonian is again written as a (negative) sum of elementary lattice interactions (Equation 1.7), the indices t and a referring to the operator “types” and lattice “units” over which the terms will be sampled. In order to represent the normalization as a sum of weights, $Z = \sum_x W(x)$, motivated by Section 1.2.1 we can insert a complete resolution of the identity between each H_{t_i, a_i} ,

$$Z = \sum_{\{\alpha\}} \sum_{S_m} \prod_{j=1}^{2m} \langle \alpha_\ell | H_{t_j, a_j} | \alpha_r \rangle. \quad (1.15)$$

We see that this has been cast in a form similar to Equation 1.8, where the sum over the set $\{\alpha\}$ and the operator list S_m must be done with importance sampling.

Note that, despite the similarity of Equation 1.15 to its finite- T analog, several important differences exist in this $T = 0$ representation. First, although the convergence parameter $2m$ is similar in magnitude to n in the “equivalent” finite- T simulation, generally speaking this projector formalism only produces meaningful observables from Equation 1.11 when the simulation is converged in m . Thus, although it is reasonable to think that the projector representation will result in a more efficient simulation for $T = 0$, in general one loses the advantage of interpreting m smaller than the converged value with some physical significance (i.e. finite temperature). Another important

⁵ Alternatively, with the imaginary-time evolution operator $e^{-\beta H}$, where β is large. The two methods are essentially equivalent however, since the exponential can be Taylor expanded.

difference is that the simulation cell has lost its periodicity in the projection (or imaginary time) direction: i.e. $|\alpha_\ell\rangle \neq |\alpha_r\rangle$ in Equation 1.15. Thus, all estimators, either in terms of the basis state, or as expectation values of an operator, are evaluated in the “middle” of the projection – i.e. a distance m from either endpoint.

In Sections 1.3 and 1.4 we will discuss the precise form of $H_{t,a}$ used to represent the Heisenberg and transverse field Ising models, as well as the specific updating schemes for sampling the basis and operator space for both finite- T and $T = 0$ QMC frameworks. However before we turn to specific models, we discuss general strategies for developing updating schemes, to be used in MCMC sampling of the $D + 1$ representations developed in the last two sections.

1.2.3 Local and non-local updating schemes

Now that we have presented two different ways of representing a D dimensional quantum lattice model by a $D + 1$ classical statistical mechanical problem, we turn very generally to the problem of how the different configurations x are sampled in a MCMC scheme. The formation of any QMC simulation corresponds to addressing two questions in the procedure of updating a configuration $x \rightarrow x'$. First, how do the transition probabilities $P(x \rightarrow x')$ depend on the weights $W(x)$ such that detailed balance,

$$W(x)P(x \rightarrow x') = W(x')P(x' \rightarrow x), \quad (1.16)$$

is satisfied. Second, how are these configuration updates performed such that the simulation is ergodic (and efficient)?

We will assume that, for sign-problem free models, an updating scheme can always be found that satisfies detailed balance (indeed, all updates are constructed as such). The question of ergodicity is more subtle. In particular, since in the SSE representation the Hamiltonian is broken up into “types”, $H_{t,a}$, different update procedures are typically needed to properly sample each operator type (e.g. diagonal or off-diagonal). These updates are roughly categorized into *local* and *non-local*, referring to the extent of the spatial or temporal region of the simulation cell that the update influences.

Local updates are familiar from classical Monte Carlo on spin systems, where the prototypical site update is simply the spin-flip \uparrow to \downarrow . These updates are possible (and recommended) in the SSE, but only at very high temperatures (or small m), on physical lattice sites that remain unconnected by operators over the entire propagation direction. Rather, the term “local” update in QMC is generally referred to as such for involving one lattice unit (e.g. a bond). Local updates are typically used to sample diagonal operators in the S^z basis; however they can be used to sample SU(2) singlet operators in the valence-bond basis (discussed below).

The term “non-local” update is also a blanket term that can refer to several different procedures. Most common are so-called “loop” updates [22],

which are essentially non-branching closed strings [23]. Loop updates have several functions: they facilitate sampling of off-diagonal operators, and also allow for fluctuations between different operators types in the simulation cell. Importantly, they are sometimes necessary for the ergodic sampling of certain estimators, such as the winding numbers, which spatially are related to quantities like the helicity modulus [24], and temporally to the uniform spin susceptibility or compressibility [25]. A variety of loop and worm algorithms have been discussed in the literature, the most important for SSE being the *directed loop* update [26]. In this review we will forgo a detailed discussion of transition probabilities in the construction of loop moves: instead we will focus entirely on so-called “deterministic” loop updates, which are available for certain Hamiltonians, discussed in the next section in the context of the Heisenberg model.

In addition to loop moves, the term “non-local” can also refer to updates in the $D + 1$ simulation cell that do not form non-branching closed loops, but share the property of being extended over a larger space-time region. The most important of these are “cluster” updates, which unlike loops are allowed to branch with many legs over the simulation cell. In Section 1.4 we will discuss the simplest type of deterministic cluster update for the transverse field Ising model [9]. Cluster updates have also been designed in S^z preserving Hamiltonians, in particular the so-called J-K models with four-site exchange [27]. In these models, although the exact implementation details are a bit more complicated, the spirit of the cluster update is the same.

Remarkably, the form of the loop or cluster update depends very little on which representation of the SSE simulation cell is used, be it $T = 0$ projector or $T > 0$ partition-function based. Rather, it depends on the specific quantum Hamiltonian one wishes to implement. In order to make our discussion more concrete, we now abandon the general discussion of the SSE in favor of making pedagogical progress on specific spin 1/2 models. In Section 1.3, we will discuss implementations of the $T = 0$ and $T > 0$ SSE for the SU(2) symmetric Heisenberg model, which is sampled efficiently with deterministic loop updates. In Section 1.4, we similarly discuss SSE implementations of the transverse field Ising model, and the associated cluster update.

1.3 Spin-1/2 Heisenberg model

Due to its important place in the history of the development of SSE, the literature describing algorithms for the spin-1/2 Heisenberg model,

$$H = \sum_{\langle ij \rangle} \mathbf{S}_i \cdot \mathbf{S}_j, \quad (1.17)$$

is extensive [1, 2, 4, 20, 23, 26]. We encourage the reader interested in implementation of the Heisenberg model to consult the appropriate authoritative

reference. Here, we discuss the basic ideas of the SSE representation, together with the important local and loop updates, as a way of grounding the discussion. We therefore focus on the unifying framework of the zero- and finite-temperature simulations, instead of all possible implementation details for the plethora of variants of this important quantum many-body model.

The standard approach to simulating the Heisenberg Hamiltonian using an SSE representation is to employ the S^z basis. Using this basis, we begin by specifying the appropriate bond-decomposition for $H_{t,a}$ (from Equation 1.7) for Equation 1.17. Namely, we use,

$$H_{0,0} = I, \quad (1.18)$$

$$H_{1,a} = \frac{1}{4} - S_i^z S_j^z, \quad (1.19)$$

$$H_{2,a} = \frac{1}{2}(S_i^+ S_j^- + S_i^- S_j^+), \quad (1.20)$$

from which matrix-element weights can be constructed to give $W(x)$. First note that, in the above bond decomposition, two facts are evident:

1. There has been a constant term $1/4$ added to the diagonal operator $H_{1,a}$. This eliminates the offensive negative sign: however it adds a constant $1/4 \times N_b$, where N_b is the number of nearest-neighbor bonds, to the expectation value of the Hamiltonian (i.e. the energy).
2. There has been a *rotation* of the spin operators by $\pi/2$ around the z -axis on one of the sublattices of a bipartite lattice. This ensures that the off-diagonal part of the Hamiltonian, $H_{2,a}$, remains positive-definite as required. Note however that this “trick” of eliminating the negative sign is only possible on bipartite lattices.⁶

The weight $W(x)$ of a sampled configuration x is proportional to the product of all matrix elements in the SSE expansion. Each matrix element is calculated using the S^z basis: representing the $S_i^z = +1/2$ eigenstate as $|\bullet\rangle_i$ and $S_j^z = -1/2$ as $|\circ\rangle_j$, the only non-zero matrix elements are,

$$\langle \bullet \circ | H_{1,a} | \bullet \circ \rangle = \langle \circ \bullet | H_{1,a} | \circ \bullet \rangle = \frac{1}{2}, \quad (1.21)$$

$$\langle \bullet \circ | H_{2,a} | \circ \bullet \rangle = \langle \circ \bullet | H_{2,a} | \bullet \circ \rangle = \frac{1}{2}. \quad (1.22)$$

We now see the choice of $1/4$ in Equation 1.19 to be judicious: the matrix elements contributing to the total weight of a configuration, $W(x)$, are equal when non-zero. This will significantly simplify sampling, we will now see for both the finite- T and $T = 0$ cases.

⁶ A generalization of this basis rotation on non-bipartite lattices would amount to a solution to the aforementioned “sign-problem” – and a likely Nobel prize for its architect.

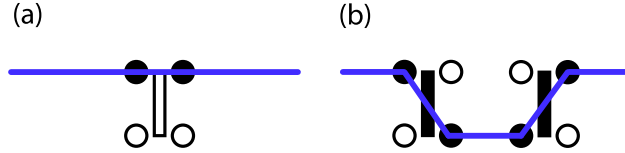


Fig. 1.1. In (a), a particle worldline (blue) that encounters a two-site diagonal operator $\langle \bullet \circ | H_{1,a} | \bullet \circ \rangle$ continues unmolested. In (b), a worldline that encounters an off-diagonal operator $\langle \bullet \circ | H_{2,a} | \circ \bullet \rangle$ is translated by one lattice unit. If worldlines are to remain periodic in imaginary time, another off-diagonal operator $\langle \circ \bullet | H_{2,a} | \bullet \circ \rangle$ is needed to translate the worldline back.

1.3.1 Finite-temperature SSE in the S^z basis

It is common to call the space-time path that a $S^z = +1/2$ state traces out in the $D+1$ dimension expansion a “worldline”. Figure 1.1 makes it evident that, in the SSE operator decomposition discussed above, the diagonal ($H_{1,a}$) and off-diagonal ($H_{2,a}$) operators affect the worldline path quite differently. Off-diagonal operators disturb it with a translation, while diagonal operators do not. Because of this, diagonal operators can be sampled in each propagation time-slice with *local* updates that replace $H_{1,a}$ on a specific lattice bond a with the null operator $H_{0,0}$, or vice versa. The transition probabilities (in the Metropolis sense) associated with these “diagonal updates” are derived directly from Equations 1.10 and 1.16, by considering the ratio of weights $W(x')/W(x)$. One finds, to add a diagonal operator (thereby changing the power in the propagator list from n to $n+1$),

$$P(n \rightarrow n+1) = \min\left(\frac{1}{2} \frac{N_b \beta}{(M-n)}, 1\right), \quad (1.23)$$

where the number of lattice bonds, N_b , enters in since the bond index a must be chosen at random for the insertion. The factor of $1/2$ comes from the matrix elements 1.21. Similarly, to remove an operator $H_{1,a}$ from the list, the transition probability is

$$P(n \rightarrow n-1) = \min\left(\frac{2(M-n+1)}{N_b \beta}, 1\right). \quad (1.24)$$

These local updates are instrumental in changing the expansion order n (necessary to sample the sum \sum_n in Equation 1.8). They also illustrate an important point: diagonal updates alone do nothing to sample *off-diagonal* operators $H_{2,a}$. For this, we must devise another updating scheme.

Antiquated methods devised to sample $H_{2,a}$ involved identifying diagonal operators that were “nearby” in the propagation direction, and updating them both to produce two off-diagonal operators such as in Figure 1.1(b) [4]. Such approaches are the equivalent of local updates: they are found to be inefficient,

as well as non-ergodic e.g. in the all-important winding number [24]. With the advent of the global *loop* move, such problems are relegated to history [23, 26].

In general, a loop move is constructed by creating a “defect” (or “head”) in the worldline configuration, and propagating that defect until it closes upon itself (i.e. finds its “tail”). The path of the defect through the space-time simulation cell passes through basis states (spins) and operators in $D + 1$. This path defines the loop, which itself is just a list of space-time lattice coordinates. A loop is said to be “flipped” if all spins on the path are flipped - which also changes the operator types associated with flipped spins (i.e. the matrix elements). This loop flipping generally accrues some change in weight. These weight changes can be absorbed into the process of creating the loop path, such that the propagation of the defect is subject to the Metropolis condition at every step (this is the *directed* loop [26]).

Quite generally, loop updates hold their power in the fact that one can typically avoid the cases where the algorithm samples “zero” (or low weight) contributions to the partition function. In some cases of higher symmetry like the Heisenberg model, simple loop algorithms can be constructed that have optimal properties in this regard. This comes from our choice of Equations 1.21 and 1.22, which forces non-zero matrix elements have equal weights. In other words, if loops are constructed to only transition between non-zero matrix elements, the entire loop flip will involve no weight change: $W(x')/W(x) = 1$. Thus, a Swendsen-Wang (SW) algorithm can be employed, were all loops can be constructed, and each flipped with probability $1/2$ [28]

In Figure 1.2 an abstraction of a finite- T SSE simulation cell for the spin- $1/2$ Heisenberg model is shown, for a $n = 6$ operator configuration. The propagation direction is horizontal – periodic boundary conditions in imaginary time are represented by arrows. Closed loops, eligible for flipping, are constructed from the paths formed by solid lines. In this case, the rules for a loop’s construction are simple: make the loop head follow a horizontal path, and when it reaches an operator (vertical bar), it turns its path in the opposite direction on the associated link. Given an operator position configuration (generally decided by the diagonal update above), the loop structure formed in this manner is fully determined – hence the name *deterministic* loops. This allows for the straightforward application of the SW algorithm to determine whether loops are flipped, as discussed above.

Note the dashed lines in Figure 1.2 highlighting one loop structure involving the bottom two lattice sites. The worldline for this configuration, which is not illustrated, would look similar to Figure 1.1(b). However, upon flipping the dashed loop, the worldline would straighten out to be entirely contained on the second-from-bottom site. By examining the structure of the other loops in this example, one can see that loop algorithms such as this are capable of making large-scale changes in the worldline configuration. Hence, among other features, loop updates are expected to vastly increase the efficiency of worldline sampling in an SSE QMC simulation.

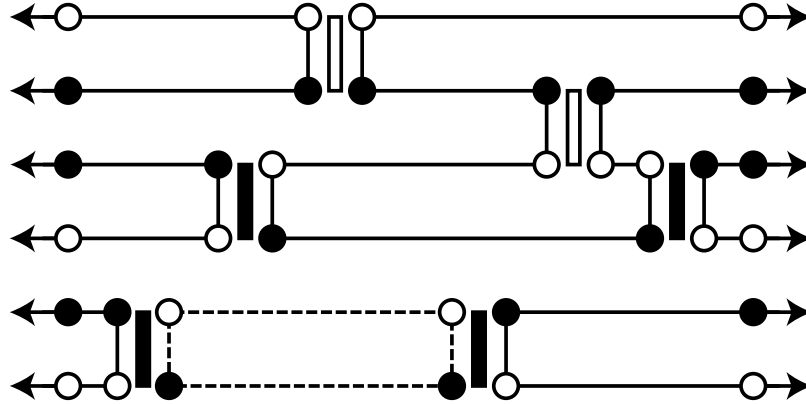


Fig. 1.2. A $D+1$ SSE simulation cell snapshot for a six-site Heisenberg model. Lattice sites are arranged vertically in one-dimension: the propagation direction (imaginary time) is horizontal, with $n = 6$ operators. Arrows represent periodic boundary conditions in the propagation direction. Solid lines are the fully-determined loop structure, where each loop can be formed and flipped with probability $1/2$. If the dashed loop is flipped, it “straightens out” the worldline segment illustrated in Figure 1.1(b), changing the off-diagonal operators (filled bars) to diagonal operators (open bars).

The interested reader can distill many of the nuances regarding implementation of the Heisenberg model with finite- T SSE from the literature. In particular, the loop updates occur within an abstracted simulation cell called a “linked list” [26]. The topology of the loop structure is fixed in each linked list; it is only the diagonal update which modifies the operator positions. Hence one again sees the necessity of both types of updates in an ergodic sampling of the Heisenberg model. The full SSE update procedure typically involves a three-step process: diagonal update, creation of the linked-list, and loop update.

We note that expectation values of physical quantities can be taken at various steps in this procedure. The simplest measurements involve quantities diagonal in the basis, such as the magnetization, that can be measured at any step in the periodic propagation direction. Expectation values of Hamiltonian operators, such as the energy, can be performed by counting the number and position of operators in the $D+1$ simulation cell; e.g.

$$E = -\frac{\langle n \rangle}{\beta}, \quad (1.25)$$

is the internal energy, up to the constant factor added to $H_{1,a}$ [23]. Other quantities are determined by the change in worldline structure, and are easiest measured in the loop update itself - for example the equal-time Green’s function, which is measured by keeping track of the position of the loop propagating defect in the $D+1$ dimensional simulation cell [19, 29, 30].

It would be possible to discuss many more interesting and important details of the SSE representation, updating, and measurement procedure, however we refer the interested reader to the literature for these. Instead, we turn to a comparison of the basic structure of the loop-algorithm SSE code, described above, to the zero-temperature formulation of the same model. As we will see, despite their formally different starting points, this comparison reveals the underlying similarity of the two methods in the SSE QMC paradigm.

1.3.2 Zero-temperature projector in the valence bond basis

The beginning point of the $T = 0$ projection, as described in Section 1.2.2, is to choose a trial state $|\alpha\rangle$, and apply a large number of Hamiltonian operators to it. For the Heisenberg (or other SU(2) Hamiltonians) a particularly convenient class of trial states are lattice coverings in the “valence bond” (VB) basis [5, 6, 31, 32], which are lists of bonds (site-pairs) covering a (bipartite) lattice. That is,

$$|\alpha\rangle = |(a_1, b_1)(a_2, b_1) \dots (a_{N/2}, b_{N/2})\rangle, \quad (1.26)$$

where each index (a, b) labels the a and b sublattice coordinate associated with a singlet state $(|\uparrow\downarrow\rangle - |\downarrow\uparrow\rangle)/\sqrt{2}$ connecting two of the N sites. This basis happens to be non-orthogonal and massively overcomplete. Also, one can see that it is suitable for sampling bond operators of the Heisenberg Hamiltonian, which can be written in the form of a singlet projection operator,

$$\frac{1}{4} - \mathbf{S}_i \cdot \mathbf{S}_j = \frac{1}{\sqrt{2}}(|\uparrow\downarrow\rangle - |\downarrow\uparrow\rangle) \frac{1}{\sqrt{2}}(\langle\uparrow\downarrow| - \langle\downarrow\uparrow|), \quad (1.27)$$

where the minus sign can again be removed by an operator rotation on bipartite lattices. With this, a $T = 0$ QMC scheme satisfying Equation 1.14 can quickly be constructed by noting that the projection operator serves to re-arrange the end-points of the valence bonds [5]. The resulting picture becomes very much like the finite-temperature SSE, where up-spins propagating from $\langle\alpha_\ell|$ to $|\alpha_r\rangle$ form “worldlines” that are re-arranged by the $2m$ operators that occur in the propagation, Equation 1.15. Indeed, the primary differences between this scheme and the finite temperature SSE are simply:

1. A fixed value of m is used in the $T = 0$ projector method, whereas n in the finite- T SSE necessarily fluctuates.
2. The $T = 0$ simulation cell is not periodic in the propagation direction; rather, particle “worldlines” terminate at the boundaries $\langle\alpha_\ell|$ and $|\alpha_r\rangle$.

Motivated by the loop algorithm of the finite- T SSE, one can then proceed with a very similar sampling scheme. First, the singlet projection operator is broken up into diagonal and off-diagonal bond operators, Equations 1.19 and 1.20. At this stage, one sees that it is advantageous to consider the simulation to be taking place in a *combined* VB-spin basis, since these operators will only give non-zero matrix elements when spin states on the respective sublattices

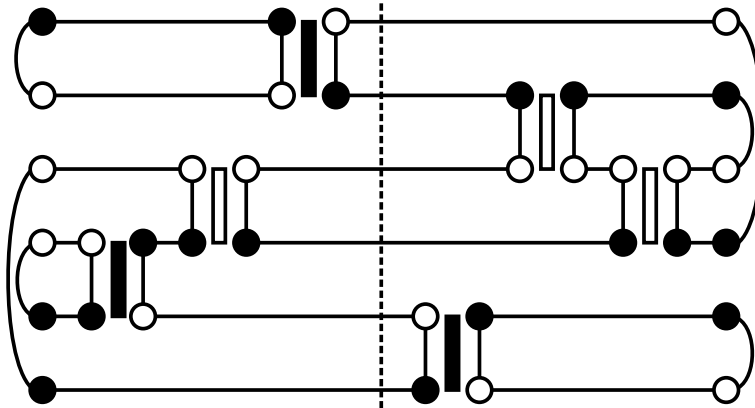


Fig. 1.3. A $D+1$ projector simulation cell snapshot for a six-site Heisenberg model. Lattice sites are arranged vertically in one-dimension: at left, spin states are $|\bullet\bullet\circ\circ\circ\bullet\rangle$, and at right, $|\circ\bullet\bullet\circ\bullet\circ\rangle$. The propagation direction is horizontal, with $m = 3$ operators. Solid lines are the fully-determined loop structure, which, when they encounter the trial states at the ends, propagate through the valence bonds (curved lines). The dashed line indicates the middle of the simulation cell, where expectation values must be measured.

are opposite [33]. As a consequence of point 2 above, we will be interested in the “middle” states of the projection to take our expectation values. These can be seen to be dependent on the particular list of bond operators for a given sample r :

$$\left(\prod_{j=1}^m H_{t_j, a_j}\right)_r |\alpha_r\rangle \propto |V_r\rangle, \quad (1.28)$$

where the proportionality constant is the numerical value for the weight of the projection.

Now, sampling can occur in analogy to the finite- T SSE with both a diagonal and a loop update. The diagonal update is significantly modified from the finite-temperature case, in part because no null ($H_{0,0}$) operators are included in the $T = 0$ projection. Then, the update proceeds by traversing the operator list $\prod_{i=1}^{2m} H_{t_i, a_i}$, and attempting the following transitions:

1. If an off-diagonal operator $H_{2,a}$ acting on some bond a is encountered, move on to the next operator, making sure to flip the associated spin state.
2. If a diagonal operator $H_{1,a}$ is present on some bond a , pick a random bond a' containing antiparallel spins, and move the operator to it.

The diagonal update being complete, one now performs the loop update, which because of the equal matrix elements of $H_{1,a}$ and $H_{2,a}$, is executed in direct analogy to the finite- T SSE case. The only difference is that loops no longer connect periodically in the propagation direction; instead, they follow

the valence bonds when they encounter the trial states $\langle\alpha_\ell|$ or $|\alpha_r\rangle$. As illustrated in Figure 1.3, the resulting schematic picture of the $D + 1$ simulation cell is, besides this lack of periodicity, essentially equivalent to Figure 1.2.

One other important difference occurs in the calculation of observables, Equation 1.11, which based on the fact that our trial states have been projected from the left and right by m operators, must be measured in the *middle* of the simulation cell. Expectation values of the form Equation 1.11 are therefore written,

$$\langle\mathcal{O}\rangle = \frac{\langle V_\ell|\mathcal{O}|V_r\rangle}{\langle V_\ell|V_r\rangle}, \quad (1.29)$$

where the states are the result of a projection from each endpoint, Equation 1.28. Remarkably, many estimators can be written as a function of the loop structure that crosses the middle of the simulation cell as in Figure 1.3 [6, 32, 33]. For example, the spin-spin correlation function $\mathbf{S}_i \cdot \mathbf{S}_j$ gives a value of 0 when i and j belong to different loops crossing the boundary, and either $\pm\frac{3}{4}$ when they're in the same loop (depending on the sublattice of each spin). The expectation value of a sum of these operators for nearest neighbor pairs i, j (i.e. Equation 1.17) can be sampled over the simulation in order to obtain an estimate for the energy. Other examples of simple-to-calculate estimators include higher-order spin correlation functions (such as dimer-dimer correlation functions), the sublattice magnetization, and even the Renyi entanglement entropy in an extended simulation cell [34]. The reader is left to scour the literature for the precise procedure for calculating the expectation value of their favorite operator; we simply note here the important result that the overlap of the two valence-bond states is given by,

$$\langle V_\ell|V_r\rangle = 2^{N_{\text{loop}} - N/2}, \quad (1.30)$$

where N_{loop} is the number of loops crossing the middle of the projection (two in Figure 1.3), and N the number of sites.

Much more discussion of the VB projector algorithm is possible, including using efficient trial states and state update, details of convergence, advanced measurement techniques, etc. However, we forgo these to continue with the broad theme of this Chapter, connecting finite- T and $T = 0$ SSE. We therefore turn next to a specific implementation of a different model, to compare and contrast with the ideas introduced in this section.

1.4 Transverse field Ising model

The transverse field Ising model (TFIM) is one of the most well-studied quantum lattice model, due to the well-known mapping to a $D + 1$ dimensional classical statistical mechanics problem, and its applicability to the field of quantum phase transitions [35]. A standard form for the Hamiltonian is,

$$H = -J \sum_{\langle i,j \rangle} \sigma_i^z \sigma_j^z - h \sum_i \sigma_i^x, \quad (1.31)$$

where σ_i is a Pauli spin operator, so that σ^z has eigenvalues ± 1 . In this equation, the first sum is over lattice *bonds*, while the second sum is over lattice *sites*. An SSE procedure for simulating the TFIM was first developed by Sandvik for the finite- T representation [9]; this was subsequently adapted to a $T = 0$ projector method, discussed in Section 1.4.2.

As with the Heisenberg model, both representations have a common starting point in the decomposition of the lattice Hamiltonian to bond (and site) operators. In this case,

$$H_{0,0} = I, \quad (1.32)$$

$$H_{-1,a} = h(\sigma_a^+ + \sigma_b^-), \quad (1.33)$$

$$H_{0,a} = h, \quad (1.34)$$

$$H_{1,a} = J(\sigma_i^z \sigma_j^z + 1). \quad (1.35)$$

Note that unlike the Heisenberg case, for the TFIM, the index a has two different meanings. Also, it is evident that some simple constants have been added to the Hamiltonian: the diagonal operator $H_{0,a}$, and also the $+1$ in Equation 1.35. The first results in matrix elements with equal weight for both site operators:

$$\langle \bullet | H_{-1,a} | \circ \rangle = \langle \circ | H_{-1,a} | \bullet \rangle = h, \quad (1.36)$$

$$\langle \bullet | H_{0,a} | \bullet \rangle = \langle \circ | H_{0,a} | \circ \rangle = h. \quad (1.37)$$

The latter ensures that the only non-zero matrix element for the bond operators are,

$$\langle \bullet \bullet | H_{1,a} | \bullet \bullet \rangle = \langle \circ \circ | H_{1,a} | \circ \circ \rangle = 2J. \quad (1.38)$$

These matrix elements form the basis of the SSE representation for the TFIM, by defining the weights $W(x)$. As for the Heisenberg model, the task is now to construct an updating procedure for QMC sampling. We will see below that a non-local update analogous to the deterministic loops will be possible; in this case however there will be branching *clusters* flipped with a SW procedure. These weights, and the updating procedure, will form a common framework for both the zero- and finite-temperature SSE procedures, outlined in the next two sections.

1.4.1 Finite-temperature SSE in the S^z basis

The finite- T simulation cell is constructed in analogy to the Heisenberg model above, with n operators (of type 1.33 to 1.35) propagating the S^z basis state, and $M - n$ null operators, 1.32. The sampling of Equation 1.10 can then be

carried out in a two-part update sequence, similar to the Heisenberg model, but with several important differences. First, we discuss the equivalent of the *diagonal* update, and second, modifications of the non-local updating scheme which give us a *cluster* algorithm.

The diagonal update proceeds in the same spirit as Section 1.3.1. One traverses the list of all M operators in the propagation in sequence. If an off-diagonal operator $H_{-1,a}$ is encountered, it is ignored (but the S^z spin associated with that site is flipped in the propagation). However, if another operator type is encountered, then a new Metropolis procedure is performed:

1. If a diagonal operator is encountered ($H_{0,a}$ or $H_{1,a}$), then it is removed with probability,

$$P = \min \left(\frac{M - n + 1}{\beta[hN + (2J)N_b]}, 1 \right), \quad (1.39)$$

where N is the number of sites, and N_b the number of bonds. One then goes to the next operator in the list (regardless of whether the removal is accepted or not).

2. If a null operator ($H_{0,0}$) is encountered, try to insert a diagonal operator with the following procedure:
 - a) First the decision of what operator type to insert is made. One chooses to insert an operator of type $H_{0,a}$ with probability,

$$P(h) = \frac{hN}{hN + (2J)N_b}, \quad (1.40)$$

or an operator of type $H_{1,a}$ with probability,

$$P(J) = \frac{(2J)N_b}{hN + (2J)N_b}. \quad (1.41)$$

Note that $P(h) + P(J) = 1$

- b) After choosing a type, accept the addition of an operator with probability,

$$P = \min \left(\frac{\beta(hN + (2J)N_b)}{M - n}, 1 \right), \quad (1.42)$$

and randomly choose an appropriate bond or site to insert it. If it is a bond operator and the chosen bond has a local spin configuration that prevents the insertion of the operator (i.e. antiparallel spins) then we do not insert the operator and consider the move failed.

It is clear that these local updates are again instrumental in changing the expansion order n , however they do not sample off-diagonal operators $H_{-1,a}$, and therefore must be combined with the use of some non-local update to produce an ergodic simulation.

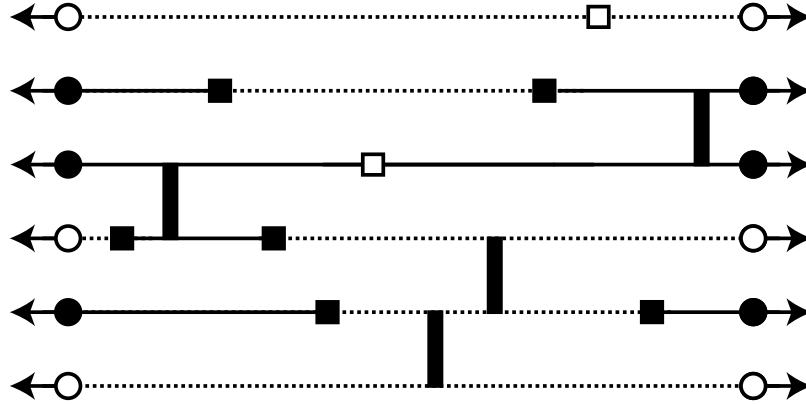


Fig. 1.4. A $D + 1$ SSE simulation cell snapshot for a six-site TFIM model. Lattice sites are arranged vertically in one-dimension: the propagation direction (imaginary time) is horizontal, with $n = 12$ operators. Vertical bars are Ising bond operators $H_{1,a}$. Filled squares are off-diagonal site operators, $H_{-1,a}$, and open squares are diagonal site operators $H_{0,a}$. Arrows represent periodic boundary conditions in the propagation direction. Solid and dashed lines illustrates the spin states (spin up or down). Flipping all the spins associated with a cluster can be done with probability $1/2$. This changes loop-terminating site operators $H_{-1,a} \leftrightarrow H_{0,a}$, but doesn't affect the bond operators or loop-internal site operators.

We require a non-local update in analogy to the loop update of the Heisenberg model above. However, in this case, a branching *cluster* update is constructed. A cluster is formed in the $D + 1$ simulation cell by grouping spins and operators together. It is built by starting e.g. with a site or bond operator, then finding the cluster that operator is a part of, according to a set of rules (Figure 1.4). The two rules governing the building of clusters is: 1) clusters terminate on site-operators $H_{-1,a}$ or $H_{0,a}$; and 2) bond operators $H_{1,a}$ belong to one cluster. This procedure continues until all clusters are identified.

The importance of including the trivial operator $H_{0,a}$ now becomes apparent, since the matrix element of both the diagonal and off-diagonal single-site operators is h . Hence, switching between $H_{0,a}$ and $H_{-1,a}$ involves no weight change. With this, clusters constructed using the algorithm can be flipped with a SW procedure, as discussed previously for the Heisenberg model.

Like the loop algorithm of the Heisenberg model, we see that the topology of the cluster structure is fixed after each diagonal update. Flipping loops has the dual effect of sampling the *type* of site operator, and also sampling the spin configuration $|\alpha\rangle$; however it is only the diagonal update which modifies the operator positions. The cluster update here is capable of making large-scale changes to the configuration of the simulation cell, however one again sees the necessity of both types of update in a fully ergodic sampling of the TFIM model.

We again refer the reader to seminal references for details of implementation, including the construction of the linked-list data structure which facilitates the practical construction of the clusters [9]. Note that measurements are again done in several different ways, either from the basis states directly (at any point in the propagation), by counting operators, or by looking at the cluster structure. We refer the reader to the relevant literature; instead, we now turn to a comparison of the cluster-algorithm TFIM code to a zero-temperature formulation of the same model.

1.4.2 Zero-temperature projector in the S^z basis

As realized recently [36], the finite-T SSE representation of the TFIM can be straightforwardly cast in the form of a $T = 0$ projector method. Unlike the Heisenberg model of Section 1.3.2, TFIM Hamiltonian operators cannot be represented as some generalized singlet projection operator, hence there is no analogous choice of basis which simplifies the sampling procedure. However, one can use a trial state which is an equal superposition of all spin states, $|\alpha\rangle = \sum_{\{\sigma^z\}} \sigma^z$, which simply involves storing a list of N spin states for the “left” and “right” trial state $|\alpha_\ell\rangle$ and $|\alpha_r\rangle$. Note that σ^z configurations in the trial state are modified during the simulation by the cluster updates, as described below.

The $D+1$ dimensional projected simulation cell is built in a way similar to Section 1.3.2, where $2m$ operators of the type 1.33 to 1.35 are sampled between the “end points” (i.e. the trial states). Recall, there are no null operators $H_{0,0}$ in the projection scheme. Then, sampling occurs via two separate procedures as previously. First, the *diagonal update* where one traverses the list of all $2m$ operators in the propagation direction in sequence. If an off-diagonal operator $H_{-1,a}$ is encountered the σ^z spin associated with that site is flipped but no operator change is made. If a diagonal operator is encountered, the Metropolis procedure is:

1. The present diagonal operator, $H_{0,a}$ or $H_{1,a}$, is removed.
2. A new operator *type* is chosen, $t = 0$ or $t = 1$, corresponding to the insertion of either a diagonal h or a diagonal J operator. The transition probability to add $H_{0,a}$ is,

$$P(h) = \frac{hN}{hN + (2J)N_b}. \quad (1.43)$$

Note, $P(J) = 1 - P(h)$ as before.

3. If $H_{0,a}$ is chosen, a site a is chosen at random, and the operator is placed there.
4. If $H_{1,a}$ is chosen, a random *bond* a is chosen. The configurations of the two spins on this bond must be parallel for the matrix element to be nonzero. If they are not, then the insertion is rejected. Since no null operators are possible, steps (2) to (4) are repeated until a successful insertion is made.

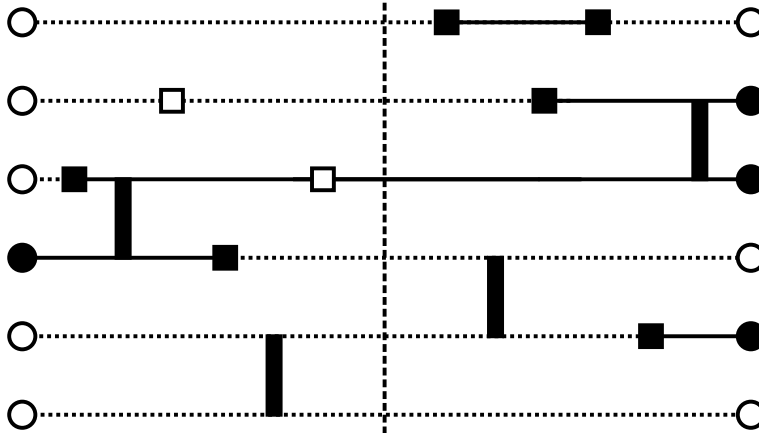


Fig. 1.5. A $D+1$ projector simulation cell snapshot for a six-site TFIM model. Lattice sites are arranged vertically in one-dimension: the propagation direction (imaginary time) is horizontal, with $m = 6$ operators. Vertical bars are Ising bond operators $H_{1,a}$. Filled squares are off-diagonal site operators, $H_{-1,a}$, and open squares are diagonal site operators $H_{0,a}$. Spin states are represented by solid or dashed lines (spin up or down). Flipping all the spins associated with each cluster can be done with probability $1/2$. This changes site operators $H_{-1,a} \leftrightarrow H_{0,a}$, and also the spin states associated with $|\alpha_\ell\rangle$ and $|\alpha_r\rangle$.

The main differences between this diagonal update and that of Section 1.4.1 is the presence of null operators in the expansion list in the latter. However, as for the finite- T SSE, one can see that this diagonal update is necessary in order to change the topology of the operator sequence in the simulation cell. In order to get fully ergodic sampling of the TFIM Hamiltonian operators, one must employ cluster updates as before.

In the $T = 0$ projection, the cluster update is carried out in almost complete analogy to the finite- T case, using a SW procedure. The main difference is that clusters terminate not only when they touch site operators, but also the spin states of the end point trial states $|\alpha_\ell\rangle$ and $|\alpha_r\rangle$. We thus see how the cluster updates sample the trial state: the flipping of a cluster that extends all the way to the simulation cell end-points will flip the σ^z states associated with the state.

Similar to Section 1.3.2, measurements are made in principle in the middle of the projection, satisfying Equation 1.11. For estimators involving basis states (e.g. the magnetization) or diagonal operators, the procedure is straightforward, and in some cases can incorporate the cluster structure at the middle of the projection. However, unlike the non-orthogonal VB basis, for off-diagonal operators, the naive expectation value can easily lead to an overlap of zero. This problem can be circumvented when operators are part of the Hamiltonian (e.g. the energy); in the interests of space however, we must leave the reader in suspense [36], or better yet, entice him or her to put down

this volume, and discover the rewarding process of devising such estimators in this versatile and transparent SSE method.

1.5 Discussion

This Chapter has aspired to give a basic flavor for the modern conceptual framework of Stochastic Series Expansion quantum Monte Carlo. Far from its first incarnations by Handscomb which were highly limited in scope, under Sandvik’s tutelage it has blossomed into a widely adopted, highly general numerical framework for solving sign-problem free quantum lattice models. Remarkably, the pace of algorithmic development ceases to abate, as evident from the large number of new systems, symmetries, measurements, and models tackled almost daily with novel manifestations of the original SSE idea. Although this Chapter focused on a pedagogical introduction to the framework of the simplest SSE simulations, the reader should not be fooled by the apparent simplicity; these methods can quickly be expanded to tackle very sophisticated quantum lattice models of real interest to modern physics. Some of the most exciting recent research on exotic quantum phases and phase transitions has employed new variations of the SSE method; notable advances have recently been reviewed elsewhere [16].

Perhaps the most important take-home message of this Chapter is the recent conceptual advances in unifying the traditional finite- T , partition function-based SSE framework, with Sandvik’s new zero-temperature projector perspective. Despite the initial apparent differences, particularly upon introduction of the valence bond projector method [5], the two paradigms have now almost completely converged, largely due to the understanding gained by considering the $T = 0$ representation and updating procedure in terms of non-local loop structures [33]. It was the purpose of this review to emphasize this fact; the reader should be struck by the similarities between Figures 1.2 and 1.3 for the Heisenberg SSE, and also Figures 1.4 and 1.5 for the TFIM, which essentially differ from each other *only* in the periodicity of the boundary condition in the propagation direction.

As we move forward as a community, we can undoubtedly expect to rely more and more on SSE QMC as a mainstay of our research on quantum lattice models. This progress will likely come with a choice of $T = 0$ or finite- T frameworks for most Hamiltonians of interest. This, coupled with innovative sampling techniques, hardware advances, and of course unforeseen algorithm breakthroughs, will ensure the golden years of quantum many-body simulation still lie ahead.

1.6 Acknowledgments

This work would not have been possible without the continuing collaboration of A. Sandvik, who is proprietor of almost every algorithmic advance outlined

in the Chapter, and without who's willingness to communicate these ideas privately would have made this work impossible. I am also indebted to A. Kallin and S. Inglis for contributions to all sections of this Chapter, especially the figures, and many critical readings.

References

1. D.C. Handscomb, Proc. Cambridge Philos. Soc. **58**, 594 (1962)
2. A.W. Sandvik, J. Kurkijärvi, Phys. Rev. B **43**, 5950 (1991)
3. A.W. Sandvik, J. Phys. A **25**, 3667 (1992)
4. A.W. Sandvik, Phys. Rev. B **56**, 11678 (1997)
5. A.W. Sandvik, Phys. Rev. Lett. **95**, 207203 (2005)
6. K.S.D. Beach, A.W. Sandvik, Nucl. Phys. B **750**, 142 (2006)
7. P. Henelius, P. Fröbrich, P.J. Kuntz, C. Timm, P.J. Jensen, Phys. Rev. B **66**, 094407 (2002)
8. R.G. Melko, J. Phys.: Condens. Matter **19**, 145203 (2007)
9. A.W. Sandvik, Phys. Rev. E **68**, 056701 (2003)
10. S. Wessel, F. Alet, M. Troyer, G.G. Batrouni, Phys. Rev. A **70**, 053615 (2004)
11. S. Wessel, F. Alet, S. Trebst, D. Leumann, M. Troyer, G.G. Batrouni, J. Phys. Soc. Jpn. Suppl. **74**, 10 (2005)
12. P. Pippan, H.G. Evertz, M. Hohenadler, Phys. Rev. A **80**, 033612 (2009)
13. K. Harada, N. Kawashima, M. Troyer, Phys. Rev. Lett. **90**(11), 117203 (2003)
14. N. Kawashima, Y. Tanabe, Phys. Rev. Lett. **98**, 057202 (2007)
15. R.K. Kaul, A.W. Sandvik, Phys. Rev. Lett. **108**, 137201 (2012)
16. R.K. Kaul, R.G. Melko, A.W. Sandvik, arXiv:1204.5405 (2012)
17. A.W. Sandvik, Phys. Rev. Lett. **104**, 177201 (2010)
18. J.S. Liu, *Monte Carlo Strategies in Scientific Computing* (Springer, 2008)
19. F.F. Assaad, H.G. Evertz, in *Computational Many-Particle Physics (Lect. Notes Phys. 739)*, ed. by H. Fehske, R. Schneider, A. Weisse (Springer Verlag, 2007), pp. 277–356
20. P. Henelius, A.W. Sandvik, Phys. Rev. B **62**, 1102 (2000)
21. N. Kawashima, K. Harada, J. Phys. Soc. Jpn. **73**, 1379 (2004)
22. H.G. Evertz, G. Lana, M. Marcu, Phys. Rev. Lett. **70**, 875 (1993)
23. A.W. Sandvik, Phys. Rev. B **59**, R14157 (1999)
24. E.L. Pollock, D.M. Ceperley, Phys. Rev. B **36**, 8343 (1987)
25. H. Evertz, Adv. Phys. **52**, 1 (2003)
26. O.F. Syljuåsen, A.W. Sandvik, Phys. Rev. E **66**, 046701 (2002)
27. R.G. Melko, A.W. Sandvik, Phys. Rev. E **72**, 026702 (2005)
28. R.H. Swendsen, J.S. Wang, Phys. Rev. Lett. **58**, 86 (1987)
29. N.V. Prokof'ev, B.V. Svistunov, I.S. Tupitsyn, Physics Letters A **238**(4-5), 253 (1998)
30. A. Dorneich, M. Troyer, Phys. Rev. E **64**, 066701 (2001)
31. S. Liang, B. Douçot, P.W. Anderson, Phys. Rev. Lett. **61**, 365 (1988)
32. A.W. Sandvik, K.S.D. Beach, in *Computer Simulation Studies in Condensed-Matter Physics XX*, ed. by D.P. Landau, S.P. Lewis, H.B. Schüttler (Springer, Berlin, 2008)
33. A.W. Sandvik, H.G. Evertz, Phys. Rev. B **82**, 024407 (2010)

34. M.B. Hastings, I. González, A.B. Kallin, R.G. Melko, Phys. Rev. Lett. **104**, 157201 (2010)
35. S. Sachdev, *Quantum Phase Transitions* (Cambridge University Press, 2011)
36. A.W. Sandvik, unpublished (2011)