Accessing the dynamics of large many-particle systems using Stochastic Series Expansion

Ansgar Dorneich

Institut für Theoretische Physik, Universität Würzburg, Am Hubland, 97074 Würzburg, Germany

Matthias Troyer

Theoretische Physik, ETH Zürich, CH-8093 Zürich, Switzerland

The Stochastic Series Expansion method (SSE) is a Quantum Monte Carlo (QMC) technique working directly in the imaginary time continuum and thus avoiding “Trotter discretization” errors. Using a non-local “operator-loop update” it allows treating large quantum mechanical systems of many thousand sites. In this paper we first give a comprehensive review on SSE and present benchmark calculations of SSE’s scaling behavior with system size and inverse temperature, and compare it to the loop algorithm, whose scaling is known to be one of the best of all QMC methods. Finally we introduce a new and efficient algorithm to measure Green’s functions and thus dynamical properties within SSE.

PACS numbers: 02.70.Ss, 05.10.Ln

I. THE SSE TECHNIQUE

Since their first formulation in the early eighties, Quantum Monte Carlo (QMC) methods have become one of the most powerful numerical simulation techniques and tools in many-body physics. The first QMC algorithms were based on a discretization in imaginary time (“Trotter decomposition”) and used purely local update steps to sample the system’s statistically relevant states. These methods require a delicate extrapolation to zero discretization in order to reduce systematic errors. Furthermore, the purely local updates often prove incapable to traverse the accessible states in an efficient way: autocorrelation times grow rapidly with increasing system size.

A more recent class of QMC algorithms, the so-called “loop algorithms”, use non-local cluster or loop update schemes, thus reducing autocorrelation times by several orders of magnitude in some cases. Unfortunately, it is often highly non-trivial to construct a loop algorithm for a new Hamiltonian, and some important interactions cannot be incorporated into the loop scheme. These interactions have to be added as a posteriori acceptance probabilities after the construction of the loop, which can seriously decrease overall efficiency of the simulation. Some loop algorithms also suffer from “freezing” when the probability is high that a certain type of cluster occupies almost the whole system.

These insufficiencies can be overcome using the “stochastic series expansion” (SSE) approach together with a loop-type updating scheme (see Ref. [14] and earlier works referenced therein).

• SSE is (almost) as efficient as loop algorithms on large systems.

• It is a numerically exact method without any discretization error.

• It is as easy to construct and general in applicability as world-line methods.

Following Sandvik [14] we briefly outline the basic ideas of SSE now. The central quantity to be sampled in a QMC simulation is the partition function

\[ Z = \text{Tr}(e^{-\beta \hat{H}}), \tag{1} \]

where \( \hat{H} \) is the system’s Hamiltonian and \( \beta = \frac{1}{T} \) the inverse temperature. Standard QMC techniques split up the exponential into a product of many “imaginary time slices” \( e^{-\Delta \tau \hat{H}} \) and truncate the Taylor expansion of this expression after a certain order in \( \Delta \tau \), thereby introducing a discretization error of order \( \Delta \tau^n \). In SSE, however, one chooses a convenient Hilbert base \( \{|\alpha\rangle\} \) (for example the \( S^z \) eigenbase \( \{|\alpha\rangle\} = \{|S_1^z, S_2^z, \ldots, S_N^z\rangle\} \)) and expands \( Z \) into the power series

\[ Z = \sum_{\alpha} \sum_{n=0}^{\infty} \frac{(-\beta)^n}{n!} \langle \alpha | \hat{H}^n | \alpha \rangle. \tag{2} \]

The statistically relevant exponents of this power series are centered around

\[ \langle n \rangle \propto N_s \beta, \tag{3} \]

where \( N_s \) is the number of sites (or orbitals) in the system. (This follows from Eq. (1) and from \( \langle E \rangle \propto N_s \).)

We can thus truncate the infinite sum over \( n \) at a finite cut-off length \( L \propto N_s \beta \) without introducing any systematic error for practical computations. The best value for \( L \) can be determined and adjusted during an initial thermalization phase of the QMC simulation: beginning with a relatively small value of \( L \) one can start the QMC update process, stop it whenever the cut-off \( L \) is exceeded and continue with \( L \) increased by \( 10\ldots20\% \).

Now let \( \hat{H} \) be composed of a certain number of elementary interactions involving one or two sites (such
as on-site potentials, nearest neighbor hopping etc.). In order to obtain a uniform notation we combine those interactions affecting only one site to new “bond” interactions. (One can, for example, take two chemical potential terms $\mu \cdot \hat{n}_{\text{site}1}$ and $\mu \cdot \hat{n}_{\text{site}2}$ and form the bond term $\frac{1}{2} \mu (\hat{n}_{\text{site}1} + \hat{n}_{\text{site}2})$ with the constant $C$ assuring that the sum over all new bond terms equals the sum over all initial on-site terms.) We can thus assume in the following that $\hat{H}$ is a finite sum of “bond” terms $\hat{H}_b$ and that the operator strings $\hat{H}^n$ in (2) can be split into terms of the form

$$\prod_{i=1}^{n} \hat{H}^{(a_i)}_b,$$

where $b_i$ labels the bond on which the elementary interaction term operates and $a_i$ the operator type (e.g. density–density interaction or hopping). By introducing “empty” unit operators $\hat{H}^{(0)} = \text{id}$ one can artificially grow all operator strings to length $L$ and obtain:

$$Z = \sum_{\alpha} \sum_{(S_L)} \frac{\beta^n (L-n)!}{L!} \langle \alpha | \prod_{i=0}^{L} (-\hat{H}^{(a_i)}_b) | \alpha \rangle. \quad \text{(5)}$$

Here $\{S_L\}$ denotes the set of all concatenations of $L$ bond operators $\hat{H}^{(a_i)}_b$ and $n$ is the number of non-unit operators in $S_L$.

If we want to sample the $(\alpha, S_L)$ according to their relative weights with a Monte Carlo procedure we have to make sure that the matrix element of each bond operator is zero or negative since in order to fulfill detailed balance we choose the acceptance probability $p$ of a bond interaction to be proportional to its negative matrix element. This requires however that all matrix elements be non-positive. Does a simple redefinition of the zero of energy help? For the diagonal operators we can indeed add the same negative constant $C$ to each of them without changing the system’s properties, and thus make all matrix elements negative or zero. Unfortunately, for the non-diagonal terms an equally simple remedy does not exist. If one can show, however, that such a non-diagonal operator must appear pairwise for the matrix element to be non-zero, its matrix element can be multiplied by $-1$ without changing the physics of the system. (This corresponds to a gauge transformation on all lattice sites with odd parity.) On non-frustrated lattices this trick is widely applicable, which considerably increases the set of Hamiltonians suitable for SSE. If there are valid world-line configurations carrying an odd number of non-diagonal vertices with positive matrix element – which is typical for Hamiltonians and lattices with frustrations – only the conventional approach of dealing with the sign problem helps: one simulates a new system with the acceptance probability $p' = |p|$ and obtains the estimate of a physical quantity $Q$ in the form

$$\langle Q \rangle = \frac{\langle Q \text{ sign } p \rangle}{\langle \text{sign } p \rangle}.$$ 

Unfortunately, $\langle \text{sign } p \rangle$ tends to zero exponentially with increasing system size $N$ and inverse temperature $\beta$, so that the computation time needed to achieve a certain accuracy exponentially increases with $N, \beta$ and the practically accessible range of system sizes and temperatures is rather limited.

II. LOOP UPDATES

Having outlined the basic idea of SSE we review the non-local updating updating scheme proposed by Sandvik. In the following figures we illustrate the proceeding by means of a simple physical model: a system of two types of hard-core bosons on a 6-site chain with periodic boundary conditions and Hamiltonian

$$H = -t \sum_{\alpha = 1, 2} \sum_i \mathcal{P} \left[ a_{\alpha,i}^\dagger a_{\alpha,i+1} + \text{H.c.} \right] + \sum_{\alpha = 1, 2} \mu_{\alpha} \sum_i n_{\alpha,i} + \sum_{\alpha = 1, 2} \eta_{\alpha} \sum_i \mathcal{P} \left[ a_{\alpha,i}^\dagger a_{\alpha,i+1} + \text{H.c.} \right]$$

with

$$\mathcal{P} = \sum_i (1 - n_{1,i} n_{2,i}). \quad \text{(6)}$$

The creation operator $a_{\alpha,i}^\dagger$ creates a hardcore boson of type $\alpha = 1$ or 2 on site $i$. The first term ($t$) is a nearest neighbor hopping term, the second term ($\mu_{\alpha}$) the chemical potential and the third term ($\eta_{\alpha}$) pair creation and annihilation. The projection operator $\mathcal{P}$ implements the hard core constraints between the two types of bosons. In the world-line representation – in which the horizontal axis represents the spatial dimension and the vertical axis the propagation level $l = 1..L$ – we symbolize type-1 bosons by single solid lines, type-2 bosons by double lines and empty sites by dotted lines (see Fig. 1).

Sandvik separates the set of all bond operators into three classes: empty operators $\hat{H}^{(0)}$, diagonal operators $\hat{H}^{(d)}$ and non-diagonal operators $\hat{H}^{(nd)}$. The QMC process starts with an arbitrarily chosen initial state $| \alpha \rangle$ and an empty operator string: in Fig. 1, for example, three sites are occupied with type-1 bosons, two sites are empty and on site 2 is occupied by a type-2 particle. Now two different update steps are performed in alternating order: a diagonal update exchanging empty and diagonal bond operators and an operator loop update transforming and exchanging diagonal and non-diagonal operators.

In the diagonal update step the operator string positions $l = 1..L$ are traversed in ascending order. If the current bond operator is a non-diagonal one it is left unchanged; if it is an empty or diagonal operator it is replaced by a diagonal or empty one with a certain probability satisfying detailed balance (i.e. an operator with
FIG. 1. world-line representation of an arbitrarily chosen start state for a physical system with three allowed occupations per site: empty (dashed line), particle 1 (solid line) or particle 2 (double line). The initial cut-off length $L$ has been set to $L = 9$, and the initial bond operator string consists only of “empty” operators.

lower energy is more likely to be maintained or inserted than an operator with a higher energy) (Fig. 2).

Following Sandvik we use the notation

$$|\alpha(l)\rangle = \prod_{i=1}^{l} \tilde{H}_{b_i}^{(\alpha_i)} |\alpha\rangle$$

for the state obtained by acting on $|\alpha\rangle$ with the first $l$ bond operators and $|\alpha_b(l)\rangle$ for the restriction of $|\alpha(l)\rangle$ to the bond $b$. Let $M$ be the total number of interacting bonds on the lattice. Then the detailed balance conditions for the diagonal update read

$$P(\tilde{H}^{(0)}(l) \rightarrow \tilde{H}_{b}^{(d)}(l)) = \min(1, \frac{M\beta \langle \alpha_b(l) | \tilde{H}_{b}^{(d)} | \alpha_b(l) \rangle}{L - n})$$

$$P(\tilde{H}_{b}^{(d)}(l) \rightarrow \tilde{H}^{(0)}(l)) = \min(1, \frac{L - n + 1}{M\beta \langle \alpha_b(l) | \tilde{H}_{b}^{(d)} | \alpha_b(l) \rangle})$$

(9)

Non-diagonal bond operators cannot simply be inserted into the world line configuration as diagonal operators can: their insertion and modification requires local changes of the world-line occupations. We discussed earlier in this paper that concatenated local changes along a closed path (or loop) through the network of world-lines and interaction vertices are much more efficient than independent purely local changes. Sandvik proposed the following method to construct such a loop: a certain world-line and a propagation level $l$ on it is chosen arbitrarily; at the chosen point one disturbs the world-line by a local change – for example the creation or annihilation of a particle. Then one chooses a direction (up or down in propagation direction) and starts moving the disturbance in this direction (Fig. 3). The aim is to move this disturbance (we will call it “loop head” in the following) through the network of world-lines and interaction vertices until the initial discontinuity is reached again and healed up.

FIG. 2. In the diagonal update step a certain number of empty bond operators is replaced by diagonal ones (and vice versa). In this example seven of the initial nine identity operators have been replaced.

FIG. 3. In the operator loop update step a local change is inserted on a world-line and then moved through the world-lines and vertices. At each vertex a new direction is chosen such that the probability of a path is proportional to the negative energy of the resulting interaction vertex (detailed balance).

Whenever the loop head reaches an interaction vertex we must decide how to go on; in the situation shown in
loop. Fortunately, all “bounce” paths can be suppressed one step backwards in the construction of the current vertex and to “bounce” backwards on the same leg. This to undo the current change on the incoming leg of the initial discontinuity is healed up.

encounterd or the starting point is reached again and the in the opposite direction until either another dead end is reached. In this case our SSE code terminates “stop here”. If this last alternative is chosen the loop has a fifth possibility to the list of allowed path choices: the by particle 2 is also a valid vertex? Then we have to add tex in Fig. 3 with three empty legs and one leg occupied it and continuing the loop. But what if the encircled ver-

ways requires choosing an outgoing leg and a change on that we have replaced 4 diagonal interactions by 4 non-
diagonal terms for particle type 2, while path “jump” is forbidden unless the Hamiltonian also allows for pair creation of particle type 2. The choice among the allowed paths must again satisfy detailed balance.

In our model – in which both pair creation and hopping are allowed – we might end up with the series “turn”, “jump”, “turn”, “turn” of path choices, after which the starting point is regained and the world-line discontinuity healed up (Fig. 4). The overall result of this loop is that we have replaced 4 diagonal interactions by 4 non-diagonal interactions (marked “n.d.”) in Fig. 4.

Sandvik’s method implicitly assumes that running with a world-line change into an interaction vertex always requires choosing an outgoing leg and a change on it and continuing the loop. But what if the encircled vertex in Fig. 4 with three empty legs and one leg occupied by particle 2 is also a valid vertex? Then we have to add a fifth possibility to the list of allowed path choices: the “stop here”. If this last alternative is chosen the loop has reached a dead end. In this case our SSE code terminates the loop here, goes back to the starting point and moves in the opposite direction until either another dead end is encountered or the starting point is reached again and the initial discontinuity is healed up.

From Fig. 3 one can see that when choosing a path through the current vertex there is always the possibility to undo the current change on the incoming leg of the vertex and to “bounce” backwards on the same leg. This path choice is normally not very helpful since it means one step backwards in the construction of the current loop. Fortunately, all “bounce” paths can be suppressed without violating detailed balance if on each bond all nonzero matrix elements are equal, or can be made equal after a suitable energy shift of the diagonal vertices. As an additional benefit, without the “bounce” path the algorithm becomes equivalent to the loop algorithms. For each vertex a path can be chosen according to detailed balance, after which the loop construction becomes deterministic. All the Heisenberg models studied in section 4 are examples for this class of “optimizable” physical systems.

A further improvement of the update scheme is possible in the limit of high temperatures, i.e. $\beta \to 0$. Equation (3) tells us that the average number of (non-empty) vertices is rather small in this situation, and a large part of all world-lines is not connected to any vertex at all. The loop update will not be very efficient here since it essentially needs a sufficient number of vertices interconnecting the world-lines. For this reason our SSE code additionally performs a so-called “free world-line update” on each world line carrying no vertex at all. In this update the occupation of the entire world-line is changed to a randomly selected new occupation.

We have stressed several times that all the local path choices satisfy detailed balance. What remains to be shown is that the updating mechanism is ergodic in the grand canonical ensemble, i.e. that all bond operator strings $S_L$ and all states $| \alpha \rangle$ can be reached. In order to demonstrate that we remind the reader that loops crossing the boundary between first and last propagation level $l$ modify the initial state $| \alpha \rangle$ for the next update cycle. Therefore, the loops sample not only $S_L$ but also $| \alpha \rangle$, and starting from a completely empty system any allowed configuration can be generated by a series of loops traversing one entire world-line each.

Numerical tests of the loop-update mechanism described above show that for large system sizes and if there are elementary interactions with very different energy scales, the loop construction sometimes gets stuck and the loop head does not find its way back to the starting point even after millions of steps. In order to avoid this trapping loops that exceed a critical length are aborted and the original state of the vertices is restored. This causes no systematic errors for measurements done between loop updates as detailed balance is not violated. The measurements of Green’s functions $G(r)$, however, which are performed “on the run” during loop construction (see Sec. 5), are biased if large loops are thrown away. Since the large loops are more likely to reach regions of the systems far away from the starting point than short loops, the values of $G(r)$ for large distances $r$ are systematically under-estimated if a considerable amount of large loops is aborted. Hence the total number of aborted loops has to be checked before one can trust in the recorded Green’s functions.
III. MEASUREMENTS

Efficient estimators for many static observables within the SSE mechanism have been derived by Sandvik [20]

- All observables \( \hat{H}(a) \) appearing as elementary interactions in the system’s Hamiltonian can be measured very easily by counting the corresponding interaction vertices in the bond operator string \( S_L \); if \( S_L \) contains on average \( \langle N(a) \rangle \) such vertices one obtains
  \[
  \langle \hat{H}(a) \rangle = -\frac{1}{\beta} \langle N(a) \rangle. \tag{10}
  \]

- Summing over all elementary terms \( \hat{H}(a) \) gives an estimator for the internal energy \( E \):
  \[
  E = \frac{1}{\beta} \langle n \rangle, \tag{11}
  \]
  where \( n \) is the number of non-empty interaction vertices in \( S_L \). (This equation can be derived very easily from \( \langle E \rangle = \frac{1}{\beta} \ln Z \).

- For the heat capacity \( C_V \) we additionally have to measure the fluctuations of \( n \):
  \[
  C_V = \langle n^2 \rangle - \langle n \rangle \tag{12}^2 - \langle n \rangle.
  \]

- Equal time correlations of two diagonal operators \( \hat{D}_1 \) and \( \hat{D}_2 \) can be measured via
  \[
  \langle \hat{D}_1 \hat{D}_2 \rangle = \left( \frac{1}{n+1} \sum_{l=0}^{n} d_2[l] d_1[l] \right), \tag{13}
  \]
  where \( d_2[l] = \langle \alpha(l) \hat{D}_1 \alpha(l) \rangle \).

Are there equally efficient estimators for time-dependent observables? In SSE the propagation index \( l \) describes the evolution of an initial state when a series of elementary terms of the Hamiltonian is acting on it; thus \( l \) plays a role analogous to imaginary time in the standard path integral. More detailed calculations show that an imaginary time separation \( \tau \) corresponds to a binomial distribution of propagation distances \( \Delta l \); the time-dependent correlation \( \langle \hat{D}_2(\tau)\hat{D}_1(0) \rangle \), for example, is related to the correlator

\[
C_{12}(\Delta l) = \frac{1}{n+1} \sum_{l=0}^{n} d_2[l + \Delta l] d_1[l] \tag{14}
\]

via

\[
\langle \hat{D}_2(\tau)\hat{D}_1(0) \rangle = \left( \sum_{\Delta l=0}^{n} \left( \frac{\tau}{\beta} \right)^{\Delta l} \left( 1 - \frac{\tau}{\beta} \right)^{n-\Delta l} C_{12}(\Delta l) \right). \tag{15}
\]

Instead of working in a representation with varying \( n \) a fixed string size \( L \) can be chosen, as the identity vertices are uniformly distributed and do not influence the mapping from index to imaginary time.

The corresponding generalized susceptibilities can be calculated straightforward by integrating \( \langle \hat{D}_2(\tau)\hat{D}_1(0) \rangle \) over \( \tau \),

\[
\chi_{12} = \int_0^\beta \langle \hat{D}_2(\tau)\hat{D}_1(0) \rangle \, d\tau. \tag{16}
\]

which gives

\[
\chi_{12} = \frac{\beta}{n(n+1)} \left( \sum_{l=0}^{n-1} d_2[l] \right) \left( \sum_{l=0}^{n-1} d_1[l] \right) + \frac{\beta}{(n+1)^2} \sum_{l=0}^{n} d_2[l] d_1[l]. \tag{17}
\]

IV. SCALING BEHAVIOR

One decisive criterion for the performance of a QMC simulation technique is the behavior of computation time \( C \) as a function of system size \( N \), or inverse temperature \( \beta \); To facilitate a hardware-independent measurement of \( C \) and a comparison to other QMC techniques we define \( C \) as the number of elementary update operations needed to transform a given state \( |\alpha(n)\rangle \) into a new state \( |\alpha(n+1)\rangle \) in such a way that the mean autocorrelation time \( \tau \) is equal to 1. In SSE the number of elementary update operations is the number of diagonal vertices tested for replacement plus the number of vertices traversed during the loop update. In the following we compare SSE to the loop-algorithm, which is known to show an excellent scaling behavior for many benchmark problems. As test models we choose isotropic antiferromagnetic Heisenberg models in one, two and three dimensions with up to 4096 sites and \( \beta \) up to 64 in a vanishing or finite external magnetic field.

Following Ref. 18 we describe the scaling behavior of the two algorithms by means of the dynamical exponent \( z \) defined from

\[
\tau \cdot C \propto \beta^{-1} l^z. \tag{18}
\]

Here, \( \tau \cdot C \) is the computational effort (i.e. the number of elementary update steps) needed to achieve a mean autocorrelation time of \( \tau = 1 \) for the measurements of the studied quantity; \( D \) is the spatial dimension of the simulated system and \( l = \sqrt{N} \) its length in each dimension. From Table I we see that both simulation techniques show an approximately equal performance and an very good scaling behavior: since the ratio \( C_\tau/(\beta N) \) is approximately constant we obtain \( z \approx 0 \) in both cases.

Next we enlarge the square lattice into the third spatial dimension and examine a bilayer quantum Heisenberg
TABLE I. 2D antiferromagnetic Heisenberg model at vanishing magnetic field $h = 0$: calculation of the uniform magnetic susceptibility $\langle \chi \rangle = \frac{\chi_{\text{tot}}}{\beta N_s}$ from QMC simulations with $1000000$ (120000 in the case $\beta = L = 64$) update-measurement cycles (left: SSE, right: loop-algorithm). $C \cdot \tau$ is the number of elementary update operations per cycle needed to achieve a mean autocorrelation time $\tau = 1$ for the measurements of $\chi$.

| $\beta \cdot N_s$ | $C \cdot \tau / \beta \cdot N_s$ | $\chi$ | $C \cdot \tau / \beta \cdot N_s$ | $\chi$ |
|------------------|-------------------------------|--------|-------------------------------|--------|
| 4 - $4^2$        | 1.00                          | 0.040(46 ± 16) | 1.00                          | 0.040(20 ± 10) |
| 8 - $8^2$        | 0.61                          | 0.044(83 ± 15) | 0.90                          | 0.044(92 ± 8)   |
| 16 - $16^2$      | 0.40                          | 0.044(72 ± 12) | 0.56                          | 0.044(68 ± 6)   |
| 32 - $32^2$      | 0.40                          | 0.044(19 ± 11) | 0.53                          | 0.044(24 ± 6)   |
| 64 - $64^2$      | 0.36                          | 0.044(01 ± 23) | 0.42                          | 0.044(07 ± 14)  |

TABLE II. Square bilayer antiferromagnetic Heisenberg model at vanishing magnetic field $h = 0$ and at the quantum critical point ($J_\perp/J = 2.524$): calculation of the uniform magnetic susceptibility $\langle \chi \rangle$ from QMC simulations with $1000000$ (390000 in the case $\beta = L = 32$) update-measurement cycles.

| $\beta \cdot N_s$ | $C \cdot \tau / \beta \cdot N_s$ | $\chi$ | $C \cdot \tau / \beta \cdot N_s$ | $\chi$ |
|------------------|-------------------------------|--------|-------------------------------|--------|
| 4 - $4^2$        | 1.00                          | 0.0115(6 ± 7)  | 1.00                          | 0.0114(6 ± 5)  |
| 8 - $8^2$        | 0.96                          | 0.0068(2 ± 6)  | 1.03                          | 0.0069(2 ± 2)  |
| 16 - $16^2$      | 0.68                          | 0.0036(8 ± 4)  | 1.20                          | 0.0036(6 ± 2)  |
| 32 - $32^2$      | 0.56                          | 0.0018(5 ± 3)  | 1.20                          | 0.0018(3 ± 1)  |

antiferromagnet at the quantum critical point separating the spin gap phase from the magnetically ordered one. Our aim is to measure scaling behavior and dynamical exponents exactly at this quantum critical point. This point is of particular interest since the immediate neighborhood of a phase transition often leads to the so-called “critical slowing down” of QMC simulations, i.e. exploding autocorrelation times and thus a dramatic decrease of efficiency of the QMC update process.

The results in Table I demonstrate that the scaling behavior for both algorithms is still almost linear in $\beta N_s$. The scaling for SSE looks slightly superior to the loop algorithm. This difference can most probably be attributed to the fact that improved estimators were used in the loop algorithm simulation, leading to slightly smaller errors but larger autocorrelation times. There is no sign of critical slowing down in either algorithm.

As we have mentioned in the introduction one of the major advantages of SSE is that external potentials (and magnetic fields in spin models) can be included without a loss of performance. To verify this assertion we now examine the antiferromagnetic Heisenberg model on a chain and a square lattice in a finite magnetic field $h \neq 0$. For the loop-algorithm we expect to find a rapidly increasing autocorrelation time and decreasing performance if the product of magnetic field $h$ and inverse temperature is much larger than 1. This is due to the fact that the external field is incorporated into the loop-algorithm via a-posteriori acceptance probabilities for each constructed loop. For $\beta h \ll 1$ these probabilities are still large, whereas at $\beta h \approx 1$ they begin to decrease considerably.

Indeed, the numerical results in Table III demonstrate that at $\beta h \approx 10$ the loop algorithm cannot be used anymore because the autocorrelation times get too long. For SSE, on the contrary, we do not expect any negative effect by introducing a magnetic field whose strength is of the order of the other elementary interactions, $h \approx J$, since no a-posteriori acceptance decision is necessary. We rather presume that performance is slightly worse for $h/J \ll 1$ because there are elementary interaction vertices with very different energy scales. Both predictions are verified by the data in Table III. For weak fields $h \ll J$ it might be preferable to construct loops in zero field, and to introduce the field via an a-posteriori Metropolis decision on whether to accept loops which change the magnetization, as it is done in the loop algorithm.

For sake of completeness we also show the corresponding data for the 2-dimensional Heisenberg model in Table I. The results demonstrate that the different behavior of SSE and loop-algorithm described in the one dimensional case is even more severe in two dimensions: the loop-algorithm loses its efficiency already at $\beta h \approx 1.5$.

In some cases other performance measurements are more interesting. One could ask how the computation time till a certain accuracy in a certain measured variable is reached scales with $\beta$ and $N_s$. This is studied in Fig. 6. For the square lattice Heisenberg antiferromagnet
we trace the computation time to reach an accuracy of 4 digits in energy as a function of $\beta$ (Fig. 5 top) and $N_s$ (Fig. 5 bottom). The exponents $\kappa(\beta)$ in $C \propto \beta^{\kappa(\beta)}$ and $\kappa(N_s)$ in $C \propto \beta^{\kappa(N_s)}$ derived from Fig. 5 are  
\[ \kappa(\beta) = 0.34 \pm 0.05, \]
\[ \kappa(N_s) = 0.48 \pm 0.05. \]

Both quantities are smaller than 1, and Eq. (18) would return a negative dynamical exponents $z$. This is due to self-averaging: in a large system local fluctuations of a physical observable around its mean value on different subregions of the lattice can compensate and average out each other, thereby lowering the observable’s measured variance. The computational effort needed to get thermodynamical averages to a certain relative error scales sublinearly with system size and inverse temperature, so that systems of several thousand sites or at temperatures of not more than 0.001$J$ can be simulated within minutes or a few hours on a standard PC or workstation.

V. GREEN’S FUNCTIONS

The observables listed in section II serve to access important static thermodynamic properties of the studied system. However, properties such as photo emission $\langle a_1^\dagger (k, \omega) a_2(0, 0) \rangle$ or spin flip $\langle S^-(k, \omega) S^+(0, 0) \rangle$ are often even more interesting as they provide insights into the system’s dynamics. Within the framework of SSE measuring these Green’s functions $G(k, \omega)$ requires the insertion of local changes on certain world-lines (such as removing a particle at propagation level $l_1$ on world-line $w_1$ and re-inserting it at propagation level $l_2$ on world-line $w_2$). Performing these insertions is a highly non-trivial task since on the one hand detailed balance must be assured, on the other hand the whole process has to sample all distances $r = w_2 - w_1$ and all propagation differences $\Delta l = l_2 - l_1$ efficiently. Both requirements are already fulfilled by the loop update steps. Since this update inserts and moves local changes on the network of world-lines and connecting interaction vertices it can be used to record the corresponding Green’s functions $G(r, \Delta l)$ “on the fly” while constructing the loop update.

As an example we reconsider the hard-core boson model from section II and in particular the operator loop shown in Figs. 3 and 4 which starts with the removal of a type-2 particle on propagation level 6 of world-line 2; our cut-off power in the series expansion was $L = 9$, and previous diagonal updates have produced $n = 7$ “non-identity” interaction vertices.

Taking level 6, the starting point of the loop, as zero point for the propagation direction we are now able to measure quantities of type $\langle a_1^\dagger (r, \Delta l) a_2(0, 0) \rangle$ and $\langle a_2^\dagger (r, \Delta l) a_2(0, 0) \rangle$ during the construction of this loop. Fig. 5 shows that for $\Delta l = 0$ exactly 2 measurements of $\langle a_2^\dagger (r, \Delta l=0) a_2(0, 0) \rangle$ and one of $\langle a_1^\dagger (r, \Delta l=0) a_2(0, 0) \rangle$ can be performed during the loop: one at the start (or end) of the loop at distance $r = 0$, two on adjacent world-lines ($r = 1$) while moving down (right) and up (left). The recorded value at each measurement is the product of the matrix elements of the creation/annihilation operators inserted at the open ends of the loop under construction. We denote the state at propagation
VI. EFFICIENTLY ACCESSING THE SYSTEM’S DYNAMICS

In this section we will discuss efficient implementation strategies for recording $G(r, \Delta l)$ and for the adjacent transformation steps mentioned above.

The transformation from propagation levels $\Delta l$ to imaginary time $\tau$ requires the same weight factors as discussed earlier for diagonal correlation functions:

$$G(r, \tau) = \sum_{\Delta l=0}^{L} \left( \frac{L}{\Delta l} \right)^{\Delta l} \left( 1 - \frac{\tau}{\beta} \right)^{L-\Delta l} G(r, \Delta l) \quad (19)$$

where

$$w(\tau, \Delta l) = \left( \frac{L}{\Delta l} \right)^{\Delta l} \left( 1 - \frac{\tau}{\beta} \right)^{L-\Delta l} . \quad (20)$$

Working in a fixed string size representation with fixed $L$ instead of varying $n$ is more convenient because the binomial weight prefactors are fixed during the entire simulation and can easily be calculated once at the beginning of the simulation.

There are several possible ways to implement the recording of $G(r, \Delta l)$ measurements and the adjacent transformation to $G(r, \tau)$. The easiest and at first glance fastest way simply writes all recorded $G(r, \Delta l)$ data into a two-dimensional array with dimensions $N_s$ and $L \times N_s \beta$. The transformation to $G(r, \tau)$ can then be performed once at the end of the simulation. However, this method has two problems. A separate measurement has to be recorded each time the loop head steps up or down by one level on a world-line and whenever it traverses an interaction vertex. Recording all these measurements drastically slows down the loop update process. Second, for large systems ($N_s \approx 5000$) and low temperatures ($\beta \approx 40$) the two-dimensional array needed to store $G(r, \Delta l)$ contains about 1 billion elements and needs more memory than available on many computer systems.

In order to overcome these problems one can replace the “brute force” recording of data on all traversed $(r, \Delta l)$ points by a Monte Carlo sampling: in each loop-update a distance $\Delta l$ is chosen randomly, according to the probabilities in Eq. (13), for each of the times $\tau$ of interest. Measurements are then performed only at these $\Delta l$ and transformed directly into $\tau$.

In our code we have adopted a third strategy: we perform all possible $G(r, \Delta l)$ measurements (thereby exploiting the fact that $G(r, \Delta l)$ is constant on the entire world-line fragment between the adjacent vertices) and directly transform these into $G(r, \tau)$ at the end of each loop update step. The transformation after each QMC update step is necessary to keep memory requirements low.
Simply applying Eq. (19) with its computationally expensive operations (divisions, powers, binomial coefficients, large sums) would now cost by far too much computation time. Instead we remember that \( G(r, \Delta l) \) is composed of a relatively small number of intervals \( I = [\Delta l_1(i), \Delta l_2(i)] \) with constant function value (Fig. 3b)). Therefore we can compute the contribution of an entire \( \Delta l \)-interval to \( G(r, \tau) \) in one step:

\[
G(r, \tau) = \sum_i G(r, I)(W(\tau, \Delta l_2(i)) - W(\tau, \Delta l_1(i)) ),
\]

where \( W \) is the “integrated weight function”

\[
W(\tau, \Delta l) = \sum_{m=0}^{\Delta l} w(\tau, m).
\]

The \( \Delta l \)-range in which \( W(\tau, \Delta l) \) considerably differs from 0 and 1 is determined by mean value and standard deviation of the binomial distribution \( w(\tau, \Delta l) \)

\[
\langle \Delta l \rangle = \frac{L}{\beta}, \quad \sigma_{\Delta l} = \sqrt{\frac{L}{\beta}(1 - \frac{\tau}{\beta})}.
\]

Below \( \langle \Delta l \rangle - 5\sigma_{\Delta l} \) the integrated weight is zero, above \( \langle \Delta l \rangle + 5\sigma_{\Delta l} \) it is 1 (up to an error of less than \( 10^{-7} \)). The remaining interval rarely contains more than fifty or hundred \( \Delta l \)-points (see Fig. 3a); these values can easily be stored after having been computed once for each \( \tau \). Thus \( W(\tau, \Delta l) \) can be calculated very rapidly with nothing but a couple of “cheap” elementary operations.

For very large systems and very low temperatures the “relevant” \( \Delta l \)-ranges might become so large that it is unfavorable to store all needed \( W(\tau, \Delta l) \) values – for example because accessing the large array \( W(\tau, \Delta l) \) would cause too many cache misses. In this case one can store the coefficients of some interpolation functions for \( W(\tau, \Delta l) \) instead of the function values themselves. Practical tests have shown that dividing the relevant interval \([\langle \Delta l \rangle - 5\sigma_{\Delta l}, \langle \Delta l \rangle + 5\sigma_{\Delta l}]\) into six sub-intervals with boundaries \([\langle \Delta l \rangle - 5\sigma_{\Delta l}, \langle \Delta l \rangle - 2.8\sigma_{\Delta l}, \langle \Delta l \rangle - 1.3\sigma_{\Delta l}, \langle \Delta l \rangle, \langle \Delta l \rangle + 1.3\sigma_{\Delta l}, \langle \Delta l \rangle + 2.8\sigma_{\Delta l}, \langle \Delta l \rangle + 5\sigma_{\Delta l}]\) and interpolating \( W \) in each sub-interval by a fifth-order polynomial is a good compromise between evaluation speed (about 15 elementary operations), storage requirements (36 floating point numbers for each \( \tau \)) and interpolation accuracy (better than \( 2.3 \times 10^{-7} \)).

The next transformation step, Fourier transform from \( G(\tau) \) to \( G(k) \), is a well known standard method that does not impose any fundamental problems. However, standard Fast Fourier Transform (FFT) algorithms perform best if all \( G(k) \) values are to be calculated, whereas in practice one rarely needs all \( k \)-values and is interested only in one \( k \)-point or in some special points of the Brillouin zone, e.g. the point \( k = (\pi, \pi) \) and its immediate neighborhood. Then one can save a lot of computation time by not recurring to FFT but using optimized algorithms designed particularly for these cases. If we are interested in only one or a few \( k \)-points we can use a simple Fourier transform to get \( \{G(k, \tau)\} \) from \( \{G(\tau, \tau)\} \) in \( \mathcal{O}(N_k \cdot n_k) \) operations (\( n_k \) is the number of \( k \)-points). Correlation functions \( C(k, \tau) \) can even be measured directly in \( k \)-space, which also can be done in \( \mathcal{O}(N_k \cdot n_k) \) operations. For the case \( 1 \ll n_k \ll N_k \) we have implemented a new Fourier transform algorithm performing much better than FFT in this situation.

Unlike a Fourier transform a Laplace transform in general cannot be inverted. Therefore the last transition step from \( \tau \) to \( \omega \) is by far more complicated than the previous one from \( \tau \) to \( k \). We use Maximum Entropy techniques developed within the last years and refer to earlier publications.
VII. EXAMPLE: SPIN CORRELATIONS OF THE 2D HEISENBERG MODEL

In this section we use our standard benchmark model – the square lattice Heisenberg antiferromagnet – to test our method of Green’s functions measurements for correctness and numerical efficiency. To this purpose we calculate and compare the correlation functions

\[ \langle S^z(\tau, k) S^z(0, k) \rangle \quad \text{and} \quad \langle S^z(\tau, k) S^-(0, k) \rangle, \tag{25} \]

\[ \langle S^z(\tau, k) S^+(0, k) \rangle \quad \text{and} \quad \langle S^+(\tau, k) S^-(0, k) \rangle. \tag{26} \]

In the $S^z$-eigenbasis, which is normally used to span the model’s Hilbert space, expression (25) is a time-dependent correlation function of two diagonal operators. Therefore the diagonal operator in Eq. (25) can be measured using Eq. (15) without introducing changes in the world-lines and vertices defining the current state of the system. The Green’s function Eq. (26), however, consists of two non-diagonal operators and can only be measured with our new method of recording general Green’s functions that was described in section V. Furthermore, at zero field $h = 0$ both correlation functions are related via

\[ \langle S^z(\tau, k) S^z(0, k) \rangle = \frac{1}{2} \langle S^z(\tau, k) S^-(0, k) \rangle, \tag{27} \]

so that the correctness of both estimators can be checked by directly comparing these two quantities.

When working with the antiferromagnet we need to keep in mind that in order to keep the exchange matrix elements $S^+ S^-$ and $S^- S^+$ positive we need to perform a gauge transformation, multiplying $S^+$ and $S^-$ on one sublattice by $-1$. This gauge transformation does not affect any diagonal operator, but leads to a momentum shift of $Q = (\pi, \pi)$, and (27) for the Green’s function and Eq. (27) becomes

\[ \langle S^z(\tau, k) S^z(0, k) \rangle = \frac{1}{2} \langle \hat{S}^z(\tau, k + Q) S^-(0, k + Q) \rangle. \tag{28} \]

The numerical data in Table V perfectly fulfill this equality and hence demonstrate the correctness of our Green’s functions measurements.

In the simulation recorded in Table V we have calculated $\langle S^z S^z \rangle$ and $\langle S^z S^- \rangle$ for all allowed $k$-points on the path $(0, 0) \to (\pi, 0) \to (\pi, \pi) \to (0, 0)$. Table V shows a subset of these points in the vicinity of $(\pi, \pi)$. The three tasks “performing updates”, “measuring $\langle S^z S^z \rangle$” and “measuring $\langle S^z S^- \rangle$” contributed the following percentages to overall computation time:

- performing updates : 18.8 %
- measuring $\langle S^z S^z \rangle$ : 36.1 %
- measuring $\langle S^+ S^- \rangle$ : 45.1 %

From this list and the measurement accuracies in Table V we conclude that the highly non-trivial Green’s functions measurements lead to a slightly better accuracy than the direct $\langle S^z(\tau, r) S^z(\tau', \tau') \rangle$ measurements while consuming roughly the same amount of computer time as the latter. Measuring the Green’s function is thus the preferred method of determining also the diagonal real space dynamical correlation functions.

VIII. SUMMARY

Stochastic Series Expansion (SSE) together with the implementation tricks and Green’s functions measurements described in this paper is a highly performant quantum Monte Carlo simulation technique allowing to access both static and dynamical properties of very large systems of thousands of sites and at very low temperatures. Compared to the loop-algorithm, which is slightly faster on big systems for some specific Hamiltonians, SSE has the advantages of not suffering from exponential slowing down in external fields; furthermore, SSE is more easily applicable to wide classes of Hamiltonians.

We thank A. Sandvik for valuable discussions. This work was supported by DFG HA 1537/16-1 and KON-WIHR OOPCV (A. D.) and the Swiss National Science Foundation (M. T.). High performance calculations were performed at HLRZ Jülich and LRZ Munich.
1 M. Suzuki, Prog. Theor. Phys. 65, 1454 (1976).
2 J. E. Hirsch, R. L. Sugar, D. J. Scalapino and R. Blankenbecler, Phys. Rev. B 26, 5033 (1982).
3 M. Takasu, S. Miyashita and M. Suzuki, Prog. Theor. Phys. 75, 1254 (1986).
4 N. Hatano, J. Phys. Soc. Japan 63, 1691 (1993).
5 H. G. Evertz, G. Lana and M. Marcu, Phys. Rev. Lett. 70, 875 (1993).
6 H.G. Evertz and M. Marcu, in Lattice 92, Amsterdam 1992, ed. J. Smit et al., Nucl. Phys. B (Proc. Suppl.) 30 (1993) 277, and in "Quantum Monte Carlo Methods in Condensed Matter Physics", ed. M. Suzuki, World Scientific 1994, p. 65.
7 U.-J. Wiese and H.-P. Ying, Phys. Lett. A168, 143 (1992); Z. Phys. B93 (1994) 147.
8 N. Kawashima, J.E. Gubernatis, and H.G. Evertz, Phys. Rev. B50 136 (1994).
9 N. Kawashima and J.E. Gubernatis, Phys. Rev. Lett. 73, 1295 (1994).
10 N. Kawashima, J. Stat. Phys. 82, 131 (1996).
11 N. Kawashima and J.E. Gubernatis, J. Stat. Phys. 80, 169 (1995).
12 H.G. Evertz, in "Numerical Methods for Lattice Quantum Many-Body Problems", ed. D.J. Scalapino, Addison Wesley Longman, Frontiers in Physics (1998).
13 M. Kohno and M. Takahashi, Phys. Rev. B 56, 3212 (1997).
14 A. W. Sandvik, Phys. Rev. B 59, R14157 (1999).
15 A. W. Sandvik and J. Kurkiäärvi, Phys. Rev. B 43, 5950 (1991); A. W. Sandvik, J. Phys. A 25, 3667 (1992).
16 A. W. Sandvik, Phys. Rev. B 56, 11678 (1997).
17 W. von der Linden, Phys. Rep. 220, 53 (1992).
18 B. Li, N. Madras and A. D. Sokal, J. Statist. Phys. 80, 661-754 (1995).
19 A. W. Sandvik and D.J. Scalapino, Phys. Rev. Lett. 72, 2777 (1994).
20 A. W. Sandvik, R. R. P. Singh and D. K. Campbell, Phys. Rev. B 56, 14510 (1997).
21 A detailed description of the new method will soon be submitted to Comp. Phys. Comm., and a C++ code package will be available from the CPC program library.
22 W. von der Linden, R. Preuss and W. Hanke, J. Phys.: Condens. Matter 8, 3881 (1996).