Poor-man's handbook for Quantum worm algorithm of Bose-Hubbard model

Ke-Hang Zhu

Contents

1 Hamiltonian for Bose Hubbard model **2**

2 Partition function **2**

3 Algorithm **5**

3.1 Subroutine: create Ira-$Masha$ pair 7

3.2 Subroutine: delete Ira-$Masha$ pair 8

3.3 Subroutine: time shift of Ira 11

3.4 Subroutine: create a kink 12

3.5 Subroutine: delete a kink 13

4 Data structure **15**

5 Quantity of interest **18**

5.1 Energy 19

5.2 Heat capacity 20

1 Hamiltonian for Bose Hubbard model

$$H = -t \sum_{\langle i,j \rangle} (b_i^\dagger b_j + b_j^\dagger b_i) + \frac{U}{2} \sum_i n_i(n_i - 1) - \mu \sum_i n_i \tag{1}$$

Here, $\langle i, j \rangle$ denotes summation over all neighboring lattice sites i and j, while $\hat{b}_i^\dagger$ and $\hat{b}_i$ are bosonic creation and annihilation operators such that $\hat{n}_i = \hat{b}_i^\dagger \hat{b}_i$ gives the number of particles on site i. The model is parametrized by the hopping amplitude t describing the mobility of bosons in the lattice, the on-site interaction U which can be attractive ($U < 0$) or repulsive ($U > 0$). The second term controls the number in each site. If $U > 0$, $\frac{U}{2} \sum_i n_i(n_i - 1)$ is the penalty for multiple bosons in one site. If $U < 0$, $\frac{U}{2} \sum_i n_i(n_i - 1)$ is the bonus for multiple bosons in one site. And the chemical potential μ , which essentially sets the total number of particles.

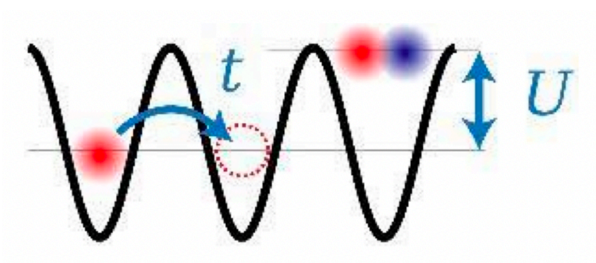

Figure 1: **Bose-Hubbard Model**

2 Partition function

$$Z = Tr(e^{-\beta H}) \tag{2}$$

Take the site's occupation number state as basis, we can decompose the

Hamiltonian into two parts:

$$H = H_0 + K \tag{3}$$

$H_0 = \frac{U}{2}\sum_i n_i(n_i-1) - \mu\sum_i n_i$ is the diagonal part in this representation, counting the chemical potential and on site energy of bosons and $K = -t\sum_{\langle i,j\rangle}(b_i^\dagger b_j + b_j^\dagger b_i)$ is the off-diagonal part, controlling the hopping strength of bosons in n.n. sites.

$$H_0 \left|\alpha\right\rangle = H_\alpha \left|\alpha\right\rangle = (\frac{U}{2}\sum_i n_i(n_i-1) - \mu\sum_i n_i)\left|\alpha\right\rangle \tag{4}$$

The transition matrix element (Hopping amplitude) could be written as

$\left\langle\gamma\left|K\right|\alpha\right\rangle$

$= \left\langle n_1, n_2, ..., n_i - 1, ...n_j + 1, ...n_N\left|K\right| n_1, n_2, ..., n_i, ...n_j, ...n_N\right\rangle$

$= K_{\gamma\alpha} = -t\sqrt{n_i(n_j+1)}$

If we perform time slicing of the partition sum,

$$Z = Tr(e^{-Hd\tau}...e^{-Hd\tau}) \tag{5}$$

$\beta = N_\tau d\tau$, N_τ is a very large number.

After inserting the complete relation into each time slice, we can get.

$$Z = \sum_{\alpha_i, \alpha_{N_\tau} = \alpha_0} \left\langle\alpha_{N_\tau}\left|e^{-d\tau H}\right|\alpha_{N_\tau-1}\right\rangle ... \left\langle\alpha_2\left|e^{-d\tau H}\right|\alpha_1\right\rangle\left\langle\alpha_1\left|e^{-d\tau H}\right|\alpha_0\right\rangle \tag{6}$$

If the state $\alpha_{i+1} = \alpha_i$ the diagonal part of the Hamiltonian contributes to the path integral

$$\left\langle\alpha_{i+1}\left|e^{-d\tau H}\right|\alpha_i\right\rangle = \left\langle\alpha_{i+1}\left|1 - d\tau H\right|\alpha_i\right\rangle = 1 - d\tau H_0(\alpha_i)\delta_{\alpha_{i+1},\alpha_i} = e^{-d\tau H_0(\alpha_i)}\delta_{\alpha_{i+1},\alpha_i} \tag{7}$$

If the state $\alpha_{i+1} \neq \alpha_i$, but can transformed into each other by one hopping, transition matrix element K contributes to the path integral would be

$$\left\langle \alpha_{i+1} \left| e^{-d\tau H} \right| \alpha_i \right\rangle = \langle \alpha_{i+1} \left| 1 - d\tau H \right| \alpha_i \rangle = -d\tau K_{\alpha_i, \alpha_{i+1}} \delta_{\alpha_{i+1}, K\alpha_i} \tag{8}$$

The delta function here means the two states can transformed into each other by K.

Plugging in these relations, we can get

$$Z = \sum_{\alpha_0, \alpha_1, .\alpha_i ..., \alpha_L \alpha_{N_\tau} = \alpha_0} e^{-d\tau H_0(\alpha_i)} \delta_{\alpha_{i+1}, \alpha_i} ...... d\tau K_{\alpha_j, \alpha_{j+1}} \delta_{\alpha_{j+1}, K\alpha_j} \tag{9}$$

Simplify the sum and the delta function in the middle, we can get the expression as follows,

$$Z = \sum_{M=0}^{\infty} \int_0^\beta d\tau_1 ... \int_{\tau_{M-1}}^\beta d\tau_M \sum_{M, \xi(M), \tau_1, ... \tau_M} \prod_{m=1}^{M} (-K_{\alpha_m \alpha_{m+1}}) exp(- \sum_{m=1}^{M} \int_{\tau_{m-1}}^{\tau_m} U_{\alpha_m} d\tau) \tag{10}$$

For convenience, I will denote $K_{\alpha_m \alpha_{m+1}}$ as $\Delta_0(\tau_m)$, meaning the transition amplitude at time τ_m.

We can write it more elegant as follows

$$Z = \sum_{M=0}^{\infty} \int_0^\beta d\tau_1 \int_{\tau_1}^\beta d\tau_2 ... \int_{\tau_{M-1}}^\beta d\tau_M \sum_{M, \xi(M), \tau_1, ... \tau_M} W(M, \xi(M), \tau_1, ... \tau_M) \tag{11}$$

The configuration uniquely described by the set of parameters $(M, \xi(M); \tau_1, ..., \tau_M)$, we need a data structure to save the information of kinks and also need to satisfy the periodic boundary condition. M is the series index and is number of hoppings (Kink) between n.n. sites. And $\tau_1, ..., \tau_M$ denote the imaginary time for these hopping to occur. ξ_M specifies the exact locations in space for all the kinks, their direction and the initial boson numbers on all sites.

With the configuration weight

$$W(M,\xi(M),\tau_1,...\tau_M) = (d\tau)^M \Delta(\tau_1)...\Delta(\tau_M) exp(-\sum_{a=1}^{M}\int_{\tau_{a-1}}^{\tau_a}(\frac{U}{2}\sum_i n_i(\tau)^2 - \tilde{\mu}\sum_i n_i(\tau))d\tau) \tag{12}$$

where $\tilde{\mu} = \mu + U/2$

If physical quantity A commutes with H_0, i.e. $[A, H_0] = 0$, its expectation value could be measured in the simulation as

$$\langle A\rangle = Z^{-1}\sum_{M=0}^{\infty}\int_0^\beta d\tau_1 \int_{\tau_1}^{\beta} d\tau_2 ... \int_{\tau_{M-1}}^{\beta} d\tau_M A(M,\xi(M),\tau_1,...,\tau_M) W(M,\xi(M),\tau_1,...,\tau_M) \tag{13}$$

In the simulation, we fix the $Masha$ end after creation of $Ira - Masha$ pair and only move Ira. Ira is the annihilation operator $b_i(\tau_{Ira})$ and can annihilate a boson at time τ_{Ira} on site i. While $Masha$ is the creation operator $b_m^\dagger(\tau_{Masha})$ and can create a boson at time τ_{Masha} on site m. The green function in imaginary time is

$$G(i,\tau_{Ira}; m, \tau_{Masha}) = \left\langle T_\tau \hat{b}_i(\tau_{Ira})\hat{b}_m^\dagger(\tau_{Masha})\right\rangle \tag{14}$$

T_τ is the Time ordering operator.

If we move Ira forward in time, the occupation number on the passed segment would increase by 1, if move Ira backward in time, the occupation number on the passed segment would decrease by 1 (turning negative is not allowed).

3 Algorithm

There are several types of suggested updates to evolve these configurations to each other.

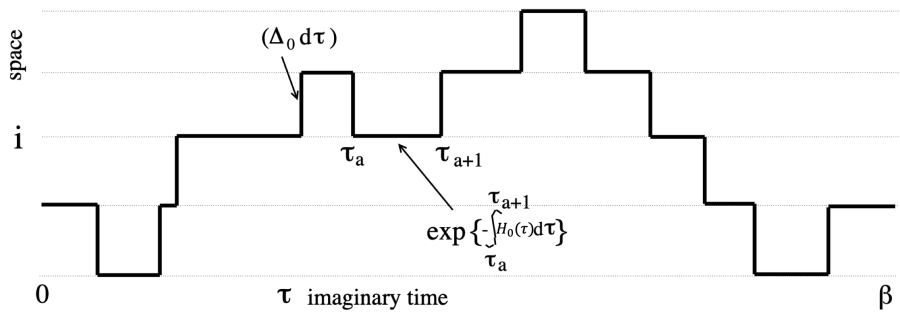

Figure 2: **Typical configuration in Z space**

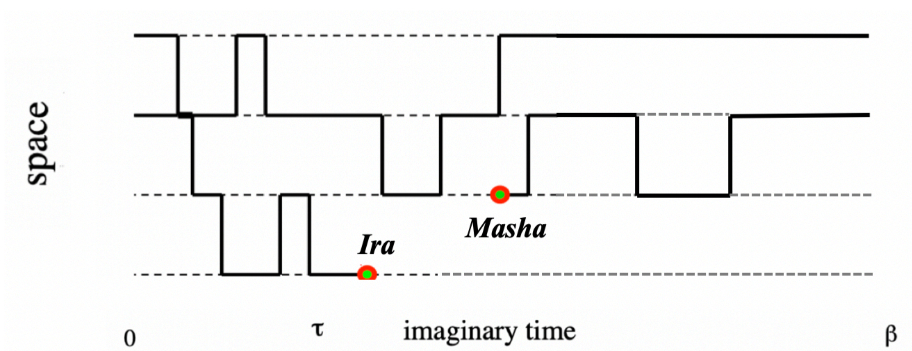

Figure 3: **Typical configuration in G space**

1. create Ira-$Masha$ pair
2. delete Ira-$Masha$ pair
3. time shift of Ira
4. create a kink
5. delete a kink

Therefore, the general algorithm would be

1. If we are in the Z configuration space, Call create Ira-$Masha$ pair subroutine
2. If we are in the G configuration space, Call possible updates with different probabilities.
   (a) p_{delw}: Call delete Ira-$Masha$ pair subroutine
   (b) p_{ts}: Call time shift subroutine
   (c) p_{crek}: Call create a kink subroutine
   (d) p_{delk}: Call delete a kink subroutine

   And the overall probability is 1, $p_{delw} + p_{ts} + p_{crek} + p_{delk} = 1$.

3.1 Subroutine: create Ira-$Masha$ pair

1. randomly select one segment out from all segments.
2. randomly choose τ_1 and τ_2 (within the period of the segment except the end points, and if τ_1 and τ_2 are the same, reject the update) to place Ira and $Masha$ respectively

3. if the segment is not a loop, suggests create two new segments with starting point being Ira and $Masha$ respectively. If the occupation number of the new segment is negantive, reject the update.

4. if the segment is a loop, choose with equal probability to act $b^\dagger$ first or act b and update the occupation number in the new segments. If the occupation number of the new segment turns negative, reject the update.

5. accept the suggested update with probability P_{acc}^{crew}

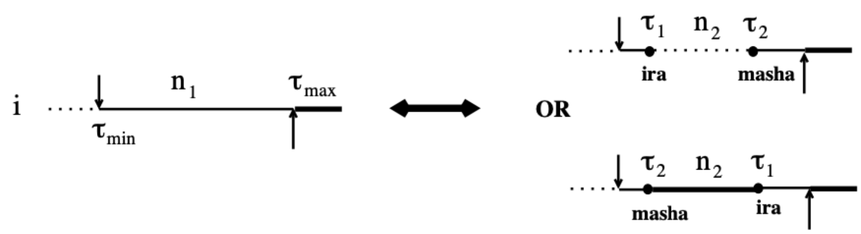

Figure 4: **Create or delete Ira-$Masha$ pair**

3.2 Subroutine: delete Ira-$Masha$ pair

Suppose the occupation number of Ira segment is $n-1$ and $Masha$ segment is n.

1. if Ira and $Masha$ segments are the only two segments composing the whole worldline, choose with equal probability to move Ira counterclockwise to annihilate $Masha$, resulting in a loop with occupation number $n-1$ and move Ira clockwise to annihilate $Masha$, resulting in a loop with occupation number n.

2. if there are other segments in the worldline, update the occupation number of the loop equals to n

3. accept the update with probability P_{acc}^{delw}

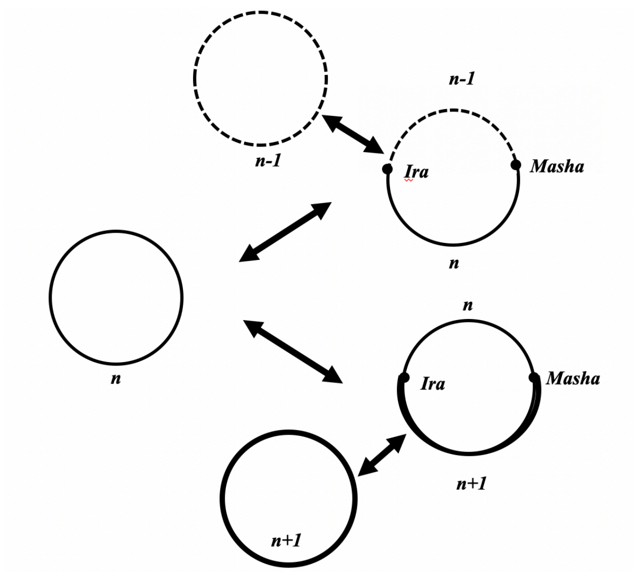

Figure 5: **Insert Ira and $Masha$ pair in a loop**

The detailed balance equation and acceptance ratio

If the segment chosen to place the Ira-$Masha$ pair is not a loop, its detailed balance equation would be

$$W_\mu (d\tau)^M \frac{1}{N_{fl}} (\tau_{max} - \tau_{min})^{-2} (d\tau)^2 P_{acc}^{\mu \to \mu'} = W_{\mu'} (d\tau)^{M+2} p_{delw} \omega_G P_{acc}^{\mu' \to \mu} \quad (15)$$

When creating Ira-$Masha$ pair, N_{fl} on the left hand side is the number of segments N_s. When deleting Ira-$Masha$ pair, because Ira-$Masha$ themselves are counted as segments, N_{fl} would be $N_s - 2$ if Ira and $Masha$ segments are not the only segment in that worldline. And N_{fl} would be $N_s - 1$ otherwise.

The probability to choose two time points τ_1 and τ_2 is $(\tau_{max} - \tau_{min})^{-2}$ because there is no sequence of τ_1 and τ_2. p_{delw} is the probability to suggest such an update. ω_G is a free parameter when we meet the transition between G and Z space and is tunable in practice. We may set it to be 1 at first and tune it to achieve best efficiency.

There are two possibilities as shown in fig4. One is $\tau_{Ira} < \tau_{Masha}$. The weight ratio of two configurations would be

$$\frac{W_{\mu'}}{W_{\mu}} = n \cdot exp(-(\tau_{Masha} - \tau_{Ira})[U((n+1)^2 - n^2)/2 - \tilde{\mu}(n+1-n)]) \tag{16}$$

n is the original occupation before inserting Ira and $Masha$.

The other possibilities is $\tau_{Ira} > \tau_{Masha}$. The weight ratio of two configurations would be

$$\frac{W_{\mu'}}{W_{\mu}} = (n+1)exp(-(\tau_{Masha} - \tau_{Ira})[U((n)^2 - (n-1)^2)/2 - \tilde{\mu}(n-(n-1))]) \tag{17}$$

In both scenarios, the acceptance would be

$$R = \frac{W_{\mu'}}{W_{\mu}} \frac{(\tau_{max} - \tau_{min})^2}{2} N_{fl} \cdot p_{delw} \cdot \omega_G \tag{18}$$

$\frac{\tau_{max} - \tau_{min}{}^2}{2}$ comes from the ordering of τ_{Ira} and τ_{Masha}.

If the segment chosen to place the Ira-$Masha$ pair is a loop, its detailed balance equation would be

$$W_{\mu}(d\tau)^M \frac{1}{N_{fl}} \frac{1}{2} (\tau_{max} - \tau_{min})^{-2} (d\tau)^2 P_{acc}^{\mu \to \mu'} = W_{\mu'} (d\tau)^{M+2} p_{delw} \frac{1}{2} \omega_G P_{acc}^{\mu' \to \mu} \tag{19}$$

The only difference between the non-loop case is that we have an additional 1/2 in each side corresponding to the two possible updates in loop case as shown in Fig5. Because they are canceled, the acceptance ratio stays the same.

3.3 Subroutine: time shift of Ira

1. access the previous segment S_1 of Ira, take its starting point as τ_{min} and the end point of the Ira segment as τ_{max}.

2. select at random the time τ_1 to move Ira within the range between τ_{max} and τ_{min}.

3. suggests to move Ira to τ_1 and update the length and endpoints of segment S_1 and Ira

4. accept the update with probability P_{acc}^{ts}

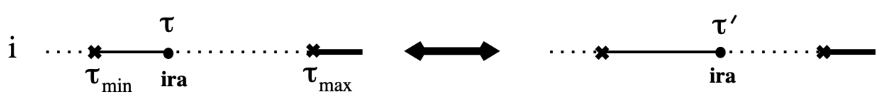

Figure 6: **Time shift**

The detailed balance equation and acceptance ratio

Suppose the occupation number of S_1 segment is n and Ira segment is $n-1$. The detailed balance equation for time shift is

$$W_{\mu}\frac{1}{\tau_{max}-\tau_{min}}P_{acc}^{\mu\to\mu'} = W_{\mu'}\frac{1}{\tau_{max}-\tau_{min}}P_{acc}^{\mu'\to\mu} \tag{20}$$

The acceptance ratio for this choice will be given by the change of the diagonal part of the Hamiltonian only

$$R = exp(-(\tau'-\tau)[U(n^2-(n-1)^2)/2-\tilde{\mu}(n-(n-1))]) \tag{21}$$

3.4 Subroutine: create a kink

1. Determine with equal probability to shift Ira forward in time or shift Ira backward in time.

2. Randomly select a site $i+u$ out from all the n.n. of the site i Ira locates. Find the adjacent segment of Ira. (the segments of $i+u$ containing the time point τ_{Ira}, If τ_{Ira} is the beginning point or the end point of the adjacent segment, reject the update to avoid the creation of segment with zero length).

3. If decides to shift Ira backward in time, compare the beginning of the segment S_1 and the beginning of the segment prior to Ira, choose the later time point to be the τ_{min} and τ_{Ira} to be τ_{max}. If decides to shift Ira forward in time, compare the endpoint of the segment S_1 and the endpoint of Ira, choose the earlier time point to be the τ_{max} and τ_{Ira} to be τ_{min}.

4. Randomly select a time point out from the range (τ_{min}, τ_{max}) to place the kink.

5. Update Ira to the same time point at site $i+u$. if after the creation, one of the occupation numbers is negative, reject the update).

6. accept the update with probability P_{acc}^{ck} to shift Ira to j_1 and place the kink at τ_1.

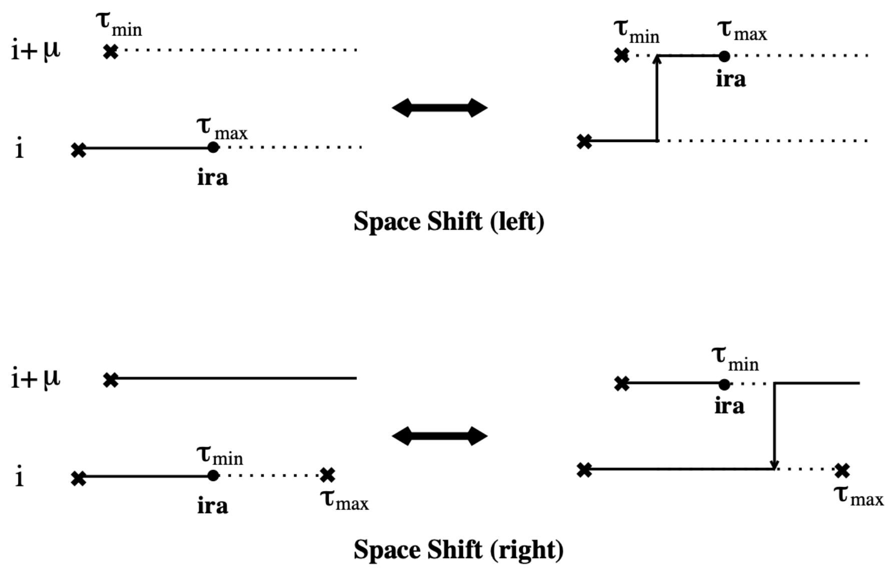

Figure 7: **Space shift**

3.5 Subroutine: delete a kink

1. Determine with equal probability to shift Ira forward in time or shift Ira backward in time.

2. Check whether the occupation number of the prior segment of Ira and the subsequent segment of Ira is the same. If not, reject the update.

3. If choose to shift backward, access the prior segment S_1 of Ira, find the segment S_2 connected by the kink on S_1. If the prior segment is not connected by any kink (masha segment), reject the update. If the ending point τ_2 of S_2 is smaller than or equals to τ_{Ira}, reject the update. Otherwise, suggests to move Ira backwards on S_1 and move through the kink and then forward on S_2 until τ_{Ira}.

4. If choose to shift forward, access the subsequent segment S_3 of Ira, find the segment S_4 connected by the first kink on S_4. If the beginning point τ_4 of S_4 is larger than or equals to τ_{Ira}, reject the update. Otherwise, suggests to move Ira forwards on S_3 and move through the kink and go backwards on S_4 until τ_{Ira}.

5. delete the kink by deleting the two segments starts from the deleted kink and update the other two segments ended up with the deleted kink. Update the occupation number according to the Ira moving law.

6. accept the update with probability P_{acc}^{dk}.

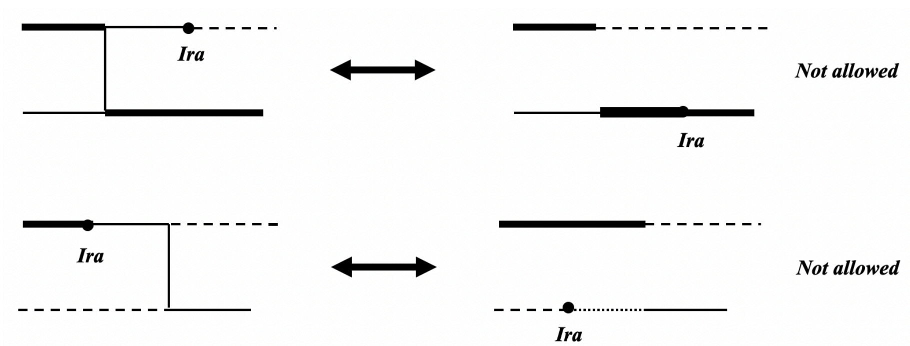

Figure 8: **Space shift which is not allowed**

The detailed balance equation and acceptance ratio

The detailed balance equation is

$$W_{\mu}(d\tau)^M \frac{1}{N_{n.n.}} \frac{1}{2} \frac{1}{\tau_{max} - \tau_{min}} d\tau p_{crek} P_{acc}^{\mu \to \mu'} = W_{\mu'}(d\tau)^{M+1} \frac{1}{2} p_{delk} P_{acc}^{\mu' \to \mu} \quad (22)$$

Here p_{crek} is the probability to suggest update of creating a kink and p_{delk} is the probability to suggest update of deleting a kink. $\frac{1}{2}$ on both sides stands for the probability of choosing the forward shift or the backward shift. $\frac{1}{N_{n.n.}}$ is the probability for selecting a nearest neighbor.

And the acceptance would be

$$R = \frac{W_{\mu'}}{W_{\mu}}(\tau_{max} - \tau_{min})N_{n.n.}\frac{p_{delk}}{p_{crek}} \tag{23}$$

When we choose to shift ira backward, the ratio between the updated configuration and original configuration would be

$$\begin{aligned}
&\frac{W_{\mu'}}{W_{\mu}} \\
&\quad = tn_2 \cdot exp(-(\tau_{max} - \tau_1)[U(n_2^2 + (n_1 - 1)^2 - (n_2 - 1)^2 - n_1^2)/2 - \tilde{\mu}(n_2 + (n_1 - 1) - (n_2 - 1) - n_1)]) \\
&\quad = tn_2 \cdot exp(-(\tau_{max} - \tau_1)[U(n2 - n1)])
\end{aligned} \tag{24}$$

When we choose to shift ira forward, the ratio between the updated configuration and original configuration would be

$$\begin{aligned}
&\frac{W_{\mu'}}{W_{\mu}} \\
&\quad = tn_2 \cdot exp(-(\tau_1 - \tau_{min})[U((n_2 - 1)^2 - n_2^2 + n_1^2 - (n_1 - 1)^2)/2 - \tilde{\mu}((n_2 - 1) - n_2 + n_1 - (n_1 - 1))]) \\
&\quad = tn_2 \cdot exp(-(\tau_1 - \tau_{min})[U(-n2 + n1)])
\end{aligned} \tag{25}$$

And as usual, $\tilde{\mu} = \mu + U/2$

4 Data structure

The worldlines of the sites can be divided into several segments by the kinks at $\tau_1 ... \tau_M$ $(M = 0, 1, ...)$ Particularly, If the worldline has no kink on it at all, the segment would be a loop with starting point 0 and length β.

The segments can be stored in a struct array. Each component in the array stores the whole information of a segment, including the site, starting point, the length of the segment, the occupation number in the segment and the segment linked by the left kink as well as the pointer to the index position in the index array. The index of the previous segment and next segment also need to be included for searching.

The struct components would be

1. site
2. starting point
3. length
4. occupation number
5. index of prior segment
6. index of subsequent segment
7. index of the segment connected by the kink
8. pointer to the index position in the index array

In order to perform the create and delete operation, we need a number array to store the index of each segment in the struct array. Their lengths are the same, but the number array would be divides into the left hand side and right hand side by a pointer separating the index of the used struct and unused struct. The used struct's index would be stored at the left hand side of the number array and null struct's index would be put on the right. When we create a segment, we can take one index number from the right hand side to the left by moving the pointer. And fill the struct of that index in the struct array. If we want to delete a segment, we can find its index in the index array and exchange it with the tail component of the left part. And move the pointer one step backward. In the struct array, we can find the according struct of that index and nullify the struct.

Ira and $Masha$ would create a segment of their own. These segments would begin with Ira or $Masha$ and ,therefore, there would be no kink information we need to store.

in order to search for the segments on the worldline of each site, we would like to store the segment index of each site crossing the imaginary time 0 point.

But in order to search for the adjacent segment defined above in the create-delete a kink subroutine, we can use the trick of super-grid. We begin by separating the whole imaginary time axis into several super-grids and store the segment indexes which cross that grids.

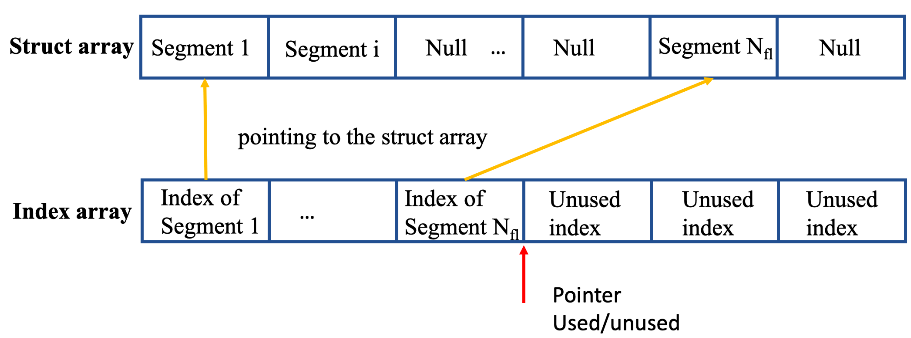

Figure 9: **Data structure to save the configuration.**

Figure 10: **Super-grid for faster search**

5 Quantity of interest

Consider an arbitrary operator A and its thermal expectation value

$$\langle A \rangle = Z^{-1} Tr(A e^{-\beta H}) \tag{26}$$

The partition function could be expressed as the sum of all configuration weights

$$Z = \sum_{\{\alpha\}} W(\{\alpha\}) \tag{27}$$

And the weight for each configuration is

$$W(\{\alpha\}) = (d\tau)^{N_K} \Delta(\tau_1)...\Delta(\tau_{N_K}) exp(-d\tau \sum_{l=0}^{N_\tau -1} H_0(l)) \tag{28}$$

Proceeding with the numerator as we did for Z, the exact time-sliced form of the expectation value can be written as

$$\langle A \rangle = Z^{-1} \sum_{\alpha} \left\langle \alpha_0 \left| A e^{-d\tau H} \right| \alpha_1 \right\rangle \left\langle \alpha_1 \left| e^{-d\tau H} \right| \alpha_2 \right\rangle ... \left\langle \alpha_{N_\tau - 1} \left| e^{-d\tau H} \right| \alpha_{N_\tau} \right\rangle \tag{29}$$

We would like to express this expectation value in the form appropriate for Monte Carlo importance sampling

$$\langle A \rangle = \frac{\sum_{\{\alpha\}} A(\{\alpha\}) W(\{\alpha\})}{\sum_{\{\alpha\}} W(\{\alpha\})} \tag{30}$$

Diagonal operators. For quantities that are diagonal in the occupation numbers, the above form is trivially valid, because $A \left|\alpha_0\right\rangle = A(\alpha_0) \left|\alpha_0\right\rangle$.

One can also average over all time slices and use

$$A(\{\alpha\}) = \frac{1}{N_\tau} \sum_{i=0}^{N_\tau - 1} A(\{\alpha_i\}) \tag{31}$$

The kinetic energy Expectation values of off-diagonal operators are in general more complicated. But if the operator is part of the Hamiltonian, we can handle it with ease. At the level of linear approximation of $e^{-d\tau H}$ in partition sum Z, we can approximate $Ke^{-d\tau H}$ in the estimator by just K. And the error is second order in $d\tau$.

Given a configuration $\{\alpha\}$ that contributes to Z, the estimator for a specific kinetic operator K_{ij} is then

$$K_{ij}(\{\alpha\}) = \frac{\langle \alpha_1 \left| K_{ij} \right| \alpha_0 \rangle}{\langle \alpha_1 \left| 1 - d\tau K_{ij} \right| \alpha_0 \rangle} \tag{32}$$

Here the numerator is non-zero only if there is a world line jump between sites i and j at the first time slice. In that case $K_{ij}(\{\alpha\}) = 1/d\tau = N_\tau/\beta$. In all other cases the estimator vanishes. We can again average over all time-slice locations of the operator K_{ij}, which results in

$$\langle K_{ij} \rangle = -\frac{\langle n_{ij} \rangle}{\beta} \tag{33}$$

where n_{ij} denotes the number of kinetic jumps in the world line configuration between sites i and j. Thus, the total kinetic energy is given by the average of the total number of kinks N_K;

$$\langle K \rangle = -\frac{\langle N_K \rangle}{\beta} \tag{34}$$

5.1 Energy

$$\langle H \rangle = \langle H_0 \rangle + \langle K \rangle \tag{35}$$

The total energy here contains the diagonal operators part and off-diagonal kinetic part. And as described above, this kind of form is very simple to handle.

We can use another method to get to formula for energy

$$\langle H \rangle = -\frac{\partial ln(Z)}{\partial \beta} = -Z^{-1}\frac{\partial Z}{\partial \beta} \tag{36}$$

$$\frac{\partial Z}{\partial \beta} = \sum_{\{\alpha\}} \frac{\partial W(\{\alpha\})}{\partial \beta} \tag{37}$$

Derivatives respect to β act on $d\tau$ in equation 28, resulting in two terms.

$$\frac{\partial W(\{\alpha\})}{\partial \beta} = \frac{N_K}{\beta} W(\{\alpha\}) + \frac{1}{\beta} \sum_{i=0}^{K} \int_{\tau_i}^{\tau_{i+1}} d\tau H_0(\tau) W(\{\alpha\}) \tag{38}$$

And we can get beck to the same formula.

$$\langle H \rangle = -\frac{\langle N_K \rangle}{\beta} + \frac{1}{\beta} \left\langle \sum_{i=0}^{K} \int_{\tau_i}^{\tau_{i+1}} d\tau H_0(\tau) \right\rangle \tag{39}$$

5.2 Heat capacity

$$C = \frac{\langle H^2 \rangle - \langle H \rangle^2}{T^2} \tag{40}$$

Since $\langle H \rangle$ is known, what we need to do is only obtaining $\langle H^2 \rangle$

$$\langle H^2 \rangle = Z^{-1} \frac{\partial^2 Z}{\partial \beta^2} \tag{41}$$

$$\langle H^2 \rangle = \frac{\langle N_K (N_K - 1) \rangle - 2 \left\langle N_K \sum_{i=0}^{K} \int_{\tau_i}^{\tau_{i+1}} d\tau H_0(\tau) \right\rangle + \left\langle (\sum_{i=0}^{K} \int_{\tau_i}^{\tau_{i+1}} d\tau H_0(\tau))^2 \right\rangle}{\beta^2} \tag{42}$$

References

[1] Prof. Youjin Deng's lecture notes