Continuous Time Quantum Monte Carlo

1 Overview

First let us separate the action into a solvable (not necessary quadratic) part and the rest $S = S_0 + \Delta S$. Partition function and average of physical quantities can be computed from

$$Z = \int \mathcal{D}[\psi^\dagger \psi] e^{-S_0 - \Delta S} \tag{1}$$

$$\langle A \rangle = \frac{1}{Z} \int \mathcal{D}[\psi^\dagger \psi] e^{-S_0 - \Delta S} A \tag{2}$$

We expand the action in power series to get the series of Feyman-like diagrams

$$Z = \sum_k \int \mathcal{D}[\psi^\dagger \psi] e^{-S_0} \frac{(-1)^k}{k!} (\Delta S)^k \tag{3}$$

Monte Carlo sampling is used to sample over all possible diagrams. (there are usually $(2k)!$ terms at order k because different order of times leads to different diagrams).

- We do not introduce time discretization, hence the name continuous time Monte Carlo.
- We do not increase the Hibert space by ising spins or other Hubbard Stratonovich fields.
- The Markov chain does not sample over ising configurations but rather over diagrams in above power series.

Most commonly we weight the diagrams according to their contribution to Z, i.e.,

$$Weight[Diagram] = \int \mathcal{D}[\psi \psi^\dagger] e^{-S_0} \frac{(-1)^k}{k!} (\Delta S)^k \tag{4}$$

The typical contribution to $(\Delta S)^k$ contains a product of $2k$ fermionic operators which are sampled.

2 A simple toy example

Lets assume that we want to sample with Monte Carlo the following expression

$$Z = e^{\int_0^\beta w(\tau)d\tau} = 1 + \int_0^\beta w(\tau)d\tau + \frac{1}{2!}\int_0^\beta d\tau_1 \int_0^\beta d\tau_2 w(\tau_1)w(\tau_2) + \cdots \quad (5)$$

If $w(\tau)$ is a function (not a quantum mechanical action), we can of course compute the integral exactly . But here we just want to see how Monte Carlo sampling can be set up if $w(\tau)$ is just a function like

$$w(\tau) = \alpha(e^{-\tau} + e^{-\beta+\tau}) \quad (6)$$

We will take the Monte Carlo probability to be proportional to its contribution to Z.

If w is a constant, we have

$$Z = e^{\int_0^\beta w(\tau)d\tau} = 1 + w\beta + \frac{1}{2!}w^2\beta^2 + \cdots + \frac{1}{k!}w^k\beta^k + \cdots \quad (7)$$

For a configuration at order k, the Monte Carlo weight should obviously be

$$P \propto \frac{\beta^k w^k}{k!} \quad (8)$$

If w is time dependent, we will have a set of times $\tau_1, \tau_2, \cdots, \tau_k$ at order k, and hence the probability should be

$$P \propto \frac{\beta^k w(\tau_1) w(\tau_2) \cdots w(\tau_k)}{k!} \tag{9}$$

What is the average perturbation order and what is probability for order k?

Lets use the notation

$$I = \int_0^\beta w(\tau) d\tau \tag{10}$$

The probability for order k is

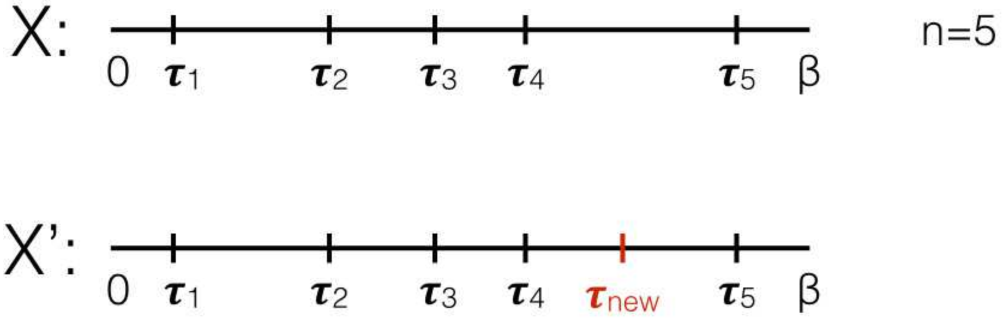
$$P_k = \frac{\frac{I^k}{k!}}{\sum_{k=0}^\infty \frac{I^k}{k!}} \tag{11}$$

The average perturbation order is

$$\langle k \rangle = \frac{\sum_{k=0}^\infty k \frac{I^k}{k!}}{\sum_{k=0}^\infty \frac{I^k}{k!}} = \frac{I \sum_{k=1}^\infty \frac{I^{k-1}}{(k-1)!}}{\sum_{k=0}^\infty \frac{I^k}{k!}} = I \tag{12}$$

Algorithm

The n-th order integral will have n-points (called kinks) in the time evolution. The MC move will either add another kink (going from order n to $n+1$), or, it will remove a kink ($n \rightarrow n-1$) from the current configuration. Here is a simple sketch:



If we are adding the new time τ_{new}, and selecting it randomly in the interval $[0, \beta]$, the probability to select the given time is τ_{new} is

$$\omega_{X_n \rightarrow X_{n+1}} = \frac{d\tau_{new}}{\beta} \tag{13}$$

This is the trial step probability for adding a kink. To keep the detail balance, we will need to equally likely removing a kink from the time evolution. We will randomly select one of the existing kinks, and removing it. In this case we have the trial step probability

$$\omega_{X_{n+1} \rightarrow X_n} = \frac{1}{n+1} \tag{14}$$

On the other hand, the probability for configurations X_n and X_{n+1} is

$$\delta\rho(X_n) = d\tau_1 d\tau_2 \cdots d\tau_n \quad w(\tau_1)w(\tau_2)\cdots w(\tau_n) \tag{15}$$

$$\delta\rho(X_{n+1}) = d\tau_1 d\tau_2 \cdots d\tau_n d\tau_{new} \quad w(\tau_1)w(\tau_2)\cdots w(\tau_n)w(\tau_{new}) \tag{16}$$

So that the acceptance step probability to add a king is well defined and is:

$$\frac{A_{X_n\rightarrow X_{n+1}}}{A_{X_{n+1}\rightarrow X_n}} = \frac{\omega_{X_{n+1}\rightarrow X_n}\ \delta\rho(X_{n+1})}{\omega_{X_n\rightarrow X_{n+1}}\ \delta\rho(X_n)} \tag{17}$$

$$= \frac{1}{n+1}\frac{\beta}{d\tau_{new}}\frac{d\tau_1 d\tau_2 \cdots d\tau_n d\tau_{new} \quad w(\tau_1)w(\tau_2)\cdots w(\tau_n)w(\tau_{new})}{d\tau_1 d\tau_2 \cdots d\tau_n \quad w(\tau_1)w(\tau_2)\cdots w(\tau_n)} \tag{18}$$

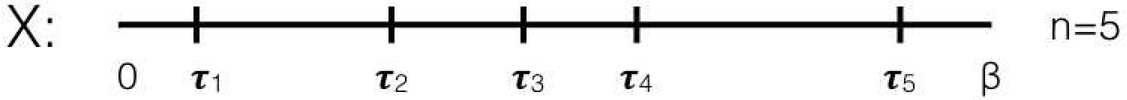
$$= \frac{\beta w(\tau_{new})}{n+1} \tag{19}$$

Similarly, to remove a kink at k (going from order $n \rightarrow n-1$) has probability

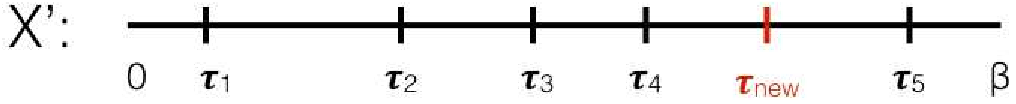
$$\frac{A_{X_n\rightarrow X_{n-1}}}{A_{X_{n-1}\rightarrow X_n}} = \frac{\omega_{X_{n-1}\rightarrow X_n}\ \delta\rho(X_{n-1})}{\omega_{X_n\rightarrow X_{n-1}}\ \delta\rho(X_n)} \tag{20}$$

$$= \frac{n}{1}\frac{d\tau_k}{\beta}\frac{d\tau_1 d\tau_2 \cdots d\tau_k \cdots d\tau_n \quad w(\tau_1)w(\tau_2)\cdots w(\tau_k)\cdots w(\tau_n)}{d\tau_1 d\tau_2 \cdots d\tau_n \quad w(\tau_1)w(\tau_2)\cdots w(\tau_n)} = \frac{n}{\beta w(\tau_k)}$$

$$\delta\rho(X) = d\tau_1 d\tau_2 \cdots d\tau_n w(\tau_1, \tau_2, \cdots, \tau_n)$$



$$\delta\rho(X') = d\tau_1 d\tau_2 \cdots d\tau_n d\tau_{new} w(\tau_1, \tau_2, \cdots, \tau_n; \tau_{new})$$



2.1 Deriving the impurity action

Skip this section if you are familiar with the quantum impurity model

The Hamiltonian for quantum impurity from previous lecture is

$$H = \sum_s \epsilon_0 c^\dagger_{0\sigma} c_{0\sigma} + U n_{0\uparrow} n_{0\downarrow} + \sum_{p>0\sigma} [V_{0p} c^\dagger_{0\sigma} c_{p\sigma} + V^*_{0p} c^\dagger_{p\sigma} c_{0\sigma}] + \sum_{p>0,\sigma} \epsilon_p c^\dagger_{p\sigma} c_{p\sigma} \tag{21}$$

It is quadratic in bath operators $c_{p>0}$ therefore they can be integrated out.

In the Feyman path integral representation we have

$$Z = \int \mathcal{D}[\psi^\dagger \psi] \mathcal{D}[c^\dagger_p c_p] e^{-S_a - \Delta S} \tag{22}$$

with atomic action

$$S_a = \int_0^\beta d\tau \left[\sum_s \psi^\dagger_\sigma(\tau)(\frac{\partial}{\partial\tau} - \mu + \epsilon_0)\psi_\sigma(\tau) + U\psi^\dagger_\uparrow(\tau)\psi_\uparrow(\tau)\psi^\dagger_\downarrow(\tau)\psi_\downarrow(\tau) \right] \tag{23}$$

and hybridization with the bath

$$\Delta S = \int_0^\beta d\tau \sum_{p\sigma} \left[c^\dagger_{p\sigma}(\frac{\partial}{\partial\tau} - \mu + \epsilon_p) c_{p\sigma} + V_{0p}\psi^\dagger_\sigma c_{p\sigma} + V^*_{0p} c^\dagger_{p\sigma}\psi_\sigma \right] \tag{24}$$

If we rewrite the second part of the acton in Matsubara representation

$$c_p(\tau) = \sqrt{T}\sum_{\omega_n} e^{-i\omega_n\tau} c_p(\omega_n), \tag{25}$$

ΔS becomes

$$\begin{aligned} \Delta S &= \sum_{p\sigma,\omega_n} [c^\dagger_{pn\sigma}(-i\omega_n - \mu + \epsilon_p) c_{pn\sigma} + V_{0p}\psi^\dagger_{n\sigma} c_{pn\sigma} + V^*_{0p} c^\dagger_{pn\sigma}\psi_{n\sigma}] & (26) \\ &= \sum_{p\sigma,\omega_n} [(c^\dagger_{pn\sigma} + \alpha^\dagger(\omega_n)\psi^\dagger_{ns})(-i\omega_n - \mu + \epsilon_p)(c_{pn\sigma} + \alpha(\omega_n)\psi_{ns}) & (27) \\ &\quad -\psi^\dagger_{ns}\alpha^\dagger(-i\omega_n - \mu + \epsilon_p)\alpha\psi_{ns}] \end{aligned}$$

Here we used shortcut notation $c_p(\omega_n) \equiv c_{pn}$. The equality between Eq. (26) and (27) is exact provided

$$\alpha(\omega_n) = -\frac{V^*_{0p}}{i\omega_n + \mu - \epsilon_p}.$$

The first term in ΔS is integrated out and gives an unimportant constant

$$\int \prod_{pn} d\overline{c}^\dagger_{pn} d\overline{c}_{pn} e^{\overline{c}^\dagger_{pn}(g^{-1})_{pn}\overline{c}_{pn}} = \det(g^{-1})$$

while the second term gives

$$\Delta S = \sum_{p\sigma,\omega_n} \psi^\dagger_{n\sigma} \frac{V^*_{0p}V_{0p}}{i\omega+\mu-\epsilon_p}\psi_{n\sigma} = \int_0^\beta \int_0^\beta d\tau d\tau' \sum_\sigma \psi^\dagger_\sigma(\tau)\Delta(\tau-\tau')\psi_\sigma(\tau') \quad (28)$$

where

$$\Delta(i\omega) = \sum_p \frac{V^*_{0p}V_{0p}}{i\omega+\mu-\epsilon_p} \quad (29)$$

is the retarded hybridization of the impurity with the bath of cunduction electrons.

The action therefore is

$$Z = \int D[\psi^\dagger\psi] e^{-S_a - \int_0^\beta d\tau \int_0^\beta d\tau' \sum_{\alpha\alpha'} \psi^\dagger_\alpha(\tau)\Delta_{\alpha\alpha'}(\tau-\tau')\psi_{\alpha'}(\tau')} \quad (30)$$

where S_a is the atomic action.

3 Details of continuous time QMC

3.1 Sampling over kinks

It is now straighforward to expand action in powers of the hybridization

$$Z = \int D[\psi^\dagger\psi] e^{-S_a} \sum_k \frac{1}{k!} \left[\sum_{\alpha\alpha'} \int_0^\beta d\tau \int_0^\beta d\tau' \psi_{\alpha'}(\tau') \psi^\dagger_\alpha(\tau) \Delta_{\alpha\alpha'}(\tau-\tau') \right]^k \quad (31)$$

Here $\Delta(\tau-\tau')$ is a real function (if no SO) and ψ's are grassman variables

We can regroup terms such that the local contribution is separated from the bath contribution

$$Z = \int D[\psi^\dagger\psi] e^{-S_a} \sum_k \frac{1}{k!} \int_0^\beta \prod_{i=1}^k d\tau_i \int_0^\beta \prod_{i=1}^k d\tau'_i \sum_{\alpha\alpha'} \prod_{i=1}^k \left[\psi_{\alpha'_i}(\tau'_i) \psi^\dagger_{\alpha_i}(\tau_i) \right] \times \prod_{i=1}^k \Delta_{\alpha_i \alpha'_i}(\tau_i, \tau'_i). \quad (32)$$

It becomes clear that partition function is a product of two terms: the average of ψ operators over the cluster degrees of freedom, and average over the bath degrees of freedom Δ.

At the perturbation order k there are $(2k)!$ terms (all possible permutations of $2k$ ψ operators).

3.2 The toy model: at most two kinks, spinless fermions

We want to demonstrated the lowest order for spinless fermions. The generalization to an arbitrary atom is trivial.

At the first order in perturbation theory, we have

$$Z = \int D[\psi^\dagger \psi] e^{-S_a} \left(1 + \int_0^\beta d\tau_s \int_0^\beta d\tau_e \psi(\tau_e) \Delta(\tau_s - \tau_e) \psi^\dagger(\tau_s) \right) \tag{33}$$

The zeroth order term is just the solution of the atomic problem. Suppose the atomic eigenstates are

$$H_a \left|0\right\rangle = E_0 \left|0\right\rangle \tag{34}$$

$$H_a \left|a\right\rangle = E_1 \left|1\right\rangle . \tag{35}$$

Then the zeroth order partition function is

$$Z = \int D[\psi^\dagger\psi] e^{-S_a} = e^{-\beta E_0} + e^{-\beta E_1} \tag{36}$$

The first order term is

$$Z_1 = \int_0^\beta d\tau_s \int_0^\beta d\tau_e \Delta(\tau_s - \tau_e) \left\{ \int D[\psi^\dagger\psi] e^{-S_a} \psi(\tau_e)\psi^\dagger(\tau_s) \right\} \tag{37}$$

The term in the brackets $\{\cdots\}$ can be evaluated in the atomic eigenbasis:

$$\begin{aligned} &\{\theta(\tau_s < \tau_e) \mathrm{Tr}(e^{-\beta H} \psi(\tau_e)\psi^\dagger(\tau_s)) \\ &-\theta(\tau_s > \tau_e) \mathrm{Tr}(e^{-\beta H} \psi^\dagger(\tau_s)\psi(\tau_e))\} \end{aligned} \tag{38}$$

or

$$\begin{aligned} &\Big\{ \theta(\tau_s < \tau_e) \sum_n \langle n| e^{-\beta H} e^{\tau_e H} \psi e^{-\tau_e H} e^{\tau_s H} \psi^\dagger e^{-\tau_s H} |n\rangle \\ &-\theta(\tau_s > \tau_e) \sum_n \langle n| e^{-\beta H} e^{\tau_s H} \psi^\dagger e^{-\tau_s H} e^{\tau_e H} \psi e^{-\tau_e H} |n\rangle \Big\} \end{aligned} \tag{39}$$

and becomes

$$\Big\{\theta(\tau_s<\tau_e)e^{-\beta E_0}e^{-(\tau_e-\tau_s)(E_1-E_0)}\langle 0|\psi|1\rangle\langle 1|\psi^\dagger|0\rangle$$
$$-\theta(\tau_s>\tau_e)e^{-\beta E_1}e^{-(\tau_s-\tau_e)(E_0-E_1)}\langle 1|\psi^\dagger|0\rangle\langle 0|\psi|1\rangle\Big\} \qquad (40)$$

We thus obtain

$$Z_1=\int_0^\beta d\tau_s\int_0^\beta d\tau_e\Delta(\tau_s-\tau_e)\Big\{\theta(\tau_s<\tau_e)e^{-\beta E_0}e^{-(\tau_e-\tau_s)(E_1-E_0)}$$
$$-\theta(\tau_s>\tau_e)e^{-\beta E_1}e^{-(\tau_s-\tau_e)(E_0-E_1)}\Big\} \qquad (41)$$

Notice that hybridization is a fermionic function, i.e., $\Delta(-\tau)=-\Delta(\beta-\tau)$ and normally is negative for positive $\tau\in[0...\beta]$ and negative for $\tau\in[-\beta...0]$. We therefore see that both terms in Z_1 are positive, which is crucial for efficient Monte Carlo sampling.

Let's check an example. Suppose that $\Delta(\tau)=\Delta\,\delta(\tau=0^-)$. This represents an atom (not real quantum impurity) as the action becomes $S=S_a-\int_0^\beta\Delta\,\psi(\tau)\psi^\dagger(\tau)$. This means that the energy of the empty state is simply decreased by $E_0\to E_0-\Delta$. The exact solution will have partition function $Z=e^{-\beta E_0+\beta\Delta}+e^{-\beta E_1}$, which can be expanded to the first order $Z=e^{-\beta E_0}+e^{-\beta E_1}+e^{-\beta E_0}\beta\Delta+\cdots$.

If we evaluate the Eq. 41, we also get

$$Z_1 = \int_0^\beta d\tau_s \Delta \left\{ e^{-\beta E_0} \right\} = \beta \Delta e^{-\beta E_0} \tag{42}$$

The single particle Green's function of the quantum impurity model can be expressed by

$$G(\tau_e - \tau_s) = -\frac{\delta \log Z}{\delta \Delta(\tau_s - \tau_e)} \tag{43}$$

This is because

$$Z = \int D[\psi^\dagger \psi] \exp\left(-S_a + \int_0^\beta d\tau_s \int_0^\beta d\tau_e \psi(\tau_e) \Delta(\tau_s - \tau_e) \psi^\dagger(\tau_s) \right) \tag{44}$$

At the first order approximation for Z, we thus obtain the zeroth order approximation for G. Taking the appropriate derivative of Eq. 41 ($G = \frac{1}{Z}\frac{\delta Z_1}{\delta \Delta}$), we get

$$\begin{aligned} G(\tau_e - \tau_s) = -\frac{1}{Z} \Big\{ & \theta(\tau_s < \tau_e) e^{-\beta E_0} e^{-(\tau_e - \tau_s)(E_1 - E_0)} \\ & -\theta(\tau_s > \tau_e) e^{-\beta E_1} e^{-(\tau_s - \tau_e)(E_0 - E_1)} \Big\} \end{aligned} \tag{45}$$

which is the lowest order approximation for the single-particle Green's function. It is known under the name *The Hubbard-I* approximation.

3.3 The toy model: at most four kinks, spinless fermions

At the second order in perturbation theory, we have

$$Z_2 = \frac{1}{2!}\int_0^\beta d\tau_{s1}\int_0^\beta d\tau_{e1}\int_0^\beta d\tau_{s2}\int_0^\beta d\tau_{e2}\Delta(\tau_{s1}-\tau_{e1})\Delta(\tau_{s2}-\tau_{e2}) \int D[\psi^\dagger\psi]e^{-S_a}\psi(\tau_{e1})\psi^\dagger(\tau_{s1})\psi(\tau_{e2})\psi^\dagger(\tau_{s2}) \qquad (46)$$

We can separate this integral into the region where $\tau_{s1} > \tau_{s2}$ and where $\tau_{s1} < \tau_{s2}$:

$$\begin{aligned} Z_2 = \frac{Z_a}{2!}\int_0^\beta d\tau_{s1}\int_0^{\tau_{s1}} d\tau_{s2}\int_0^\beta d\tau_{e1}\int_0^\beta d\tau_{e2}\Delta(\tau_{s1}-\tau_{e1})\Delta(\tau_{s2}-\tau_{e2}) \\ \langle T_\tau\psi(\tau_{e1})\psi^\dagger(\tau_{s1})\psi(\tau_{e2})\psi^\dagger(\tau_{s2})\rangle_a \\ +\frac{Z_a}{2!}\int_0^\beta d\tau_{s1}\int_{\tau_{s1}}^\beta d\tau_{s2}\int_0^\beta d\tau_{e1}\int_0^\beta d\tau_{e2}\Delta(\tau_{s1}-\tau_{e1})\Delta(\tau_{s2}-\tau_{e2}) \\ \langle T_\tau\psi(\tau_{e1})\psi^\dagger(\tau_{s1})\psi(\tau_{e2})\psi^\dagger(\tau_{s2})\rangle_a \end{aligned} \qquad (47)$$

In the second term we will exchange τ_{s1} and τ_{s2} labels and we also realize that $\int_0^\beta d\tau_{s2}\int_{\tau_{s2}}^\beta d\tau_{s1} = \int_0^\beta d\tau_{s1}\int_0^{\tau_{s1}} d\tau_{s2}$, because both integrals correspond to the same $\int_0^\beta d\tau_{s1}\int_0^\beta d\tau_{s2}\theta(\tau_{s1} > \tau_{s2})$. We thus have

$$\begin{aligned}
Z_2 = \frac{Z_a}{2!}\int_0^\beta d\tau_{s1}\int_0^{\tau_{s1}} d\tau_{s2}\int_0^\beta d\tau_{e1}\int_0^\beta d\tau_{e2}\Delta(\tau_{s1}-\tau_{e1})\Delta(\tau_{s2}-\tau_{e2})\\
\langle T_\tau\psi(\tau_{e1})\psi^\dagger(\tau_{s1})\psi(\tau_{e2})\psi^\dagger(\tau_{s2})\rangle_a\\
+\frac{Z_a}{2!}\int_0^\beta d\tau_{s1}\int_0^{\tau_{s1}} d\tau_{s2}\int_0^\beta d\tau_{e1}\int_0^\beta d\tau_{e2}\Delta(\tau_{s2}-\tau_{e1})\Delta(\tau_{s1}-\tau_{e2})\\
\langle T_\tau\psi(\tau_{e1})\psi^\dagger(\tau_{s2})\psi(\tau_{e2})\psi^\dagger(\tau_{s1})\rangle_a \qquad (48)
\end{aligned}$$

Now we realize that $\psi^\dagger(\tau_{s2})$ and $\psi^\dagger(\tau_{s1})$ can be exchanged in the second term to make the two products equal, and since these are fermionic operators, we will get a minus sign, i.e.,

$$Z_2 = \frac{Z_a}{2!}\int_0^\beta d\tau_{s1}\int_0^{\tau_{s1}} d\tau_{s2}\int_0^\beta d\tau_{e1}\int_0^\beta d\tau_{e2} \qquad (49)$$

$$(\Delta(\tau_{s1}-\tau_{e1})\Delta(\tau_{s2}-\tau_{e2}) - \Delta(\tau_{s2}-\tau_{e1})\Delta(\tau_{s1}-\tau_{e2}))$$

$$\langle T_\tau\psi(\tau_{e1})\psi^\dagger(\tau_{s1})\psi(\tau_{e2})\psi^\dagger(\tau_{s2})\rangle_a \qquad (50)$$

or

$$Z_2 = \frac{Z_a}{2!}\int_0^\beta d\tau_{s1}\int_0^{\tau_{s1}} d\tau_{s2}\int_0^\beta d\tau_{e1}\int_0^\beta d\tau_{e2} \tag{51}$$

$$\text{Det}\begin{pmatrix} \Delta(\tau_{s1}-\tau_{e1}) & \Delta(\tau_{s1}-\tau_{e2}) \\ \Delta(\tau_{s2}-\tau_{e1}) & \Delta(\tau_{s2}-\tau_{e2}) \end{pmatrix}$$

$$\langle T_\tau \psi(\tau_{e1})\psi^\dagger(\tau_{s1})\psi(\tau_{e2})\psi^\dagger(\tau_{s2})\rangle_a \tag{52}$$

We just time-ordered creation operators. If we do the same with anhilation operators, we would get

$$Z_2 = Z_a\int_0^\beta d\tau_{s1}\int_0^{\tau_{s1}} d\tau_{s2}\int_0^\beta d\tau_{e1}\int_0^{\tau_{e1}} d\tau_{e2} \tag{53}$$

$$\text{Det}\begin{pmatrix} \Delta(\tau_{s1}-\tau_{e1}) & \Delta(\tau_{s1}-\tau_{e2}) \\ \Delta(\tau_{s2}-\tau_{e1}) & \Delta(\tau_{s2}-\tau_{e2}) \end{pmatrix}$$

$$\langle T_\tau \psi(\tau_{e1})\psi^\dagger(\tau_{s1})\psi(\tau_{e2})\psi^\dagger(\tau_{s2})\rangle_a \tag{54}$$

Notice that time ordering is still enforced in the actual evaluation of the matrix elements, ($\langle T_\tau \psi(\tau_{e1})\psi^\dagger(\tau_{s1})\psi(\tau_{e2})\psi^\dagger(\tau_{s2})\rangle_a$), hence we could equivalently write

$$Z_2 = \frac{Z_a}{(2!)^2}\int_0^\beta d\tau_{s1}\int_0^\beta d\tau_{s2}\int_0^\beta d\tau_{e1}\int_0^\beta d\tau_{e2} \tag{55}$$

$$\text{Det}\begin{pmatrix} \Delta(\tau_{s1}-\tau_{e1}) & \Delta(\tau_{s1}-\tau_{e2}) \\ \Delta(\tau_{s2}-\tau_{e1}) & \Delta(\tau_{s2}-\tau_{e2}) \end{pmatrix}$$

$$\langle T_\tau \psi(\tau_{e1})\psi^\dagger(\tau_{s1})\psi(\tau_{e2})\psi^\dagger(\tau_{s2})\rangle_a \tag{56}$$

Hence to simulate such integrals there are two ways to proceed. We can allow all times τ to be unrestricted $[0,\beta]$ and have prefactor $1/(k!)^2$, or alternatively, work with sorted times for anhilation operators (and creation operators) and remove the $1/(k!)^2$ prefactor.

Notice that such grouping of terms is particularly useful among operators which are of the same type (spin $\uparrow$ for example) because we have only a single matrix element to compute out of $(k!)^2$. Notice that such matrix elements in which τ_{s1} is exchanged by τ_{e1} for example, have no relation to each other, and therefore we still need to compute of the order $(2k)!/(k!)^2$ time ordered matrix elements.

When different spin-orbitals are involved, the determinants of hybridization have no-off diagonal elements, hence we can simplify them as products of determinants for $\uparrow$ and $\downarrow$ spins (Det$_\uparrow(\Delta)$Det$_\downarrow(\Delta)$). But notice that single matrix element would represent only $(k_\uparrow!)^2(k_\downarrow!)^2$ terms, while the number of possible time ordered matrix elements is $((2k_\uparrow + 2k_\downarrow)!)$.

3.4 Grouping terms into determinant

To avoid/reduce the ”fermionic minus sign problem”, it is important to group together of the order of $k!$ terms (at the perturbation order k) to take care of the time order of $\psi^\dagger$ (or ψ) operators. After such regrouping, one can avoid the sign problem in many important cases, such as complex atom (including Hunds coupling) in medium. Because the bath is non-interacting, one can expect that such grouping in possible. Let’s dive into it.

One way to understand that, is to notice that the generated terms in Eq. 32 correspond to Feynman-diagrams in the representation of slave-particles, where each slave particle projects the impurity to one of the atomic eigenstates.

It is clear that when for all pairs of τ_i, τ_i' we have $\tau_i < \tau_i'$ (or $\tau_i > \tau_i'$) the Feyman diagram has no crossing of the propagators. These terms correspond to the so called ”Non-crossing approximation” for the Anderson impurity model (there are also terms with not strictly increasing or decreasing time order which do not lead to crossings either). Exchanging the time order of kinks, leads to crossing of propagators. It is well known that crossing diagrams have any sign while non-crossing diagrams have all the same sign and are typically larger than the crossing terms.

In another words, by grouping into determinants one adds to any non-crossing diagram all the corresponding non-crossing diagrams, hence substantially reducing the sign problem.

We first copy Eq. 32 and use the notation

$$\langle A\rangle_{local} = \frac{1}{Z_a}\int D[\psi^\dagger\psi]e^{-S_a}A \tag{57}$$

$$Z_a = \int D[\psi^\dagger\psi]e^{-S_a} \tag{58}$$

to get

$$\begin{aligned} Z = Z_a \sum_k \frac{1}{k!}\int_0^\beta d\tau_1 \int_0^\beta d\tau_1' \cdots \int_0^\beta d\tau_k \int_0^\beta d\tau_k' \sum_{\alpha_1\alpha_1',\cdots,\alpha_k,\alpha_k'} \\ \langle \psi_{\alpha_1'}(\tau_1')\psi^\dagger_{\alpha_1}(\tau_1)\cdots\psi_{\alpha_k'}(\tau_k')\psi^\dagger_{\alpha_k}(\tau_k)\rangle_{local} \\ \times \prod_{i=1}^k \Delta_{\alpha_i\alpha_i'}(\tau_i,\tau_i'). \end{aligned} \tag{59}$$

Next we combine crossing and non-crossing diagrams into a determinant, which can be mathematically expressed by

$$
\begin{aligned}
Z \;=\;& Z_a \sum_k \frac{1}{k!}\int_0^\beta d\tau_1 \int_0^\beta d\tau_1' \cdots \int_0^\beta d\tau_k \int_0^\beta d\tau_k' \sum_{\alpha_1\alpha_1',\cdots,\alpha_k,\alpha_k'} \\
& \times \langle T_\tau \psi_{\alpha_1'}(\tau_1')\psi^\dagger_{\alpha_1}(\tau_1)\cdots\psi_{\alpha_k'}(\tau_k')\psi^\dagger_{\alpha_k}(\tau_k)\rangle_{local} \times \\
& \times \frac{1}{k!} Det \begin{pmatrix}
\Delta_{\alpha_1\alpha_1'}(\tau_1,\tau_1') & \Delta_{\alpha_1\alpha_2'}(\tau_1,\tau_2') & \cdots & \cdots \\
\cdots & \cdots & \cdots & \cdots \\
\cdots & \cdots & \cdots & \cdots \\
\Delta_{\alpha_k\alpha_1'}(\tau_k,\tau_1') & \cdots & \cdots & \Delta_{\alpha_k\alpha_k'}(\tau_k,\tau_k')
\end{pmatrix}
\end{aligned}
\tag{60}
$$

Here $Z_a = \int D[\psi^\dagger\psi]e^{-S_a}$ and average is defined by $\langle O\rangle_{local} = \frac{1}{Z_a}\int D[\psi^\dagger\psi]e^{-S_a}O$.

This is the central equation of the Continuous Time Monte Carlo sampling around the atomic limit. To derive Eq. (60) from Eq. (59) one needs to permute time integration variables τ in all possible ways. Permutation of fermions gives minus sign in an odd permutation. This minus sign can be absorbed in the minus sign of the product of hybridizations, resulting in the determinant of hybridizations.

Any good quantum number of the atomic Hamiltonian can be used to simplify the large determinant into product of determinants. For example, if the spin is preserved, we would have a product of determinants for up and down spin. All matrix elements in which the equivalent operators are exchanged, are identical, and hence can be compute all at once.

3.4.1 Discussion in grouping and relation to Feynman diagrams

Let's take a look at the concrete example. The lowest nontrivial term is the second order term of the form

$$\psi(\tau_1')\psi^\dagger(\tau_1)\psi^\dagger(\tau_2)\psi(\tau_2')\Delta(\tau_1-\tau_1')\Delta(\tau_2-\tau_2'). \tag{61}$$

which is time ordered when $\tau_1' < \tau_1 < \tau_2 < \tau_2'$. We discussed the same product above. When τ_1 is bigger than τ_2, we get a contribution

$$-\psi(\tau_1')\psi^\dagger(\tau_2)\psi^\dagger(\tau_1)\psi(\tau_2')\Delta(\tau_1-\tau_1')\Delta(\tau_2-\tau_2'). \tag{62}$$

If we change the integration variable $\tau_1 \leftrightarrow \tau_2$, we have

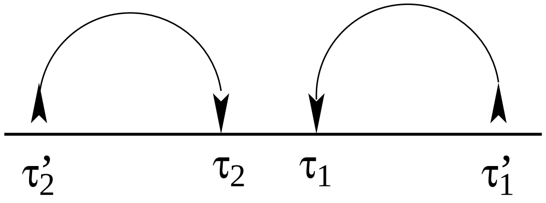
$$\psi(\tau_1')\psi^\dagger(\tau_1)\psi^\dagger(\tau_2)\psi(\tau_2')[-\Delta(\tau_2-\tau_1')\Delta(\tau_1-\tau_2')]. \tag{63}$$

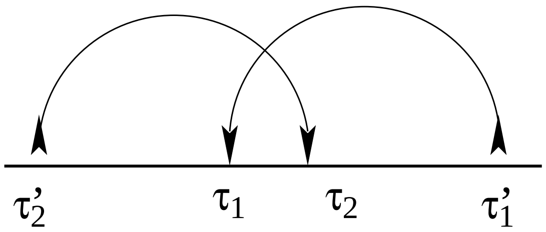
The two terms can be groped together

$$\int_0^\beta \cdots \int_0^\beta d\tau_i d\tau_i' \psi(\tau_1')\psi^\dagger(\tau_1)\psi^\dagger(\tau_2)\psi(\tau_2')\Delta(\tau_1-\tau_1')\Delta(\tau_2-\tau_2') = \tag{64}$$

$$\int_0^\beta \cdots \int_0^\beta d\tau_i d\tau_i' \psi(\tau_1')\psi^\dagger(\tau_1)\psi^\dagger(\tau_2)\psi(\tau_2')\frac{1}{2} Det \begin{pmatrix} \Delta(\tau_1-\tau_1') & \Delta(\tau_1-\tau_2') \\ \Delta(\tau_2'-\tau_1) & \Delta(\tau_2-\tau_2') \end{pmatrix}$$

Diagrammatically, the two terms can be picture in the folowing way:





3.4.2 Algorithm

Sampling of the above diagrams (60) could be done in the following way:

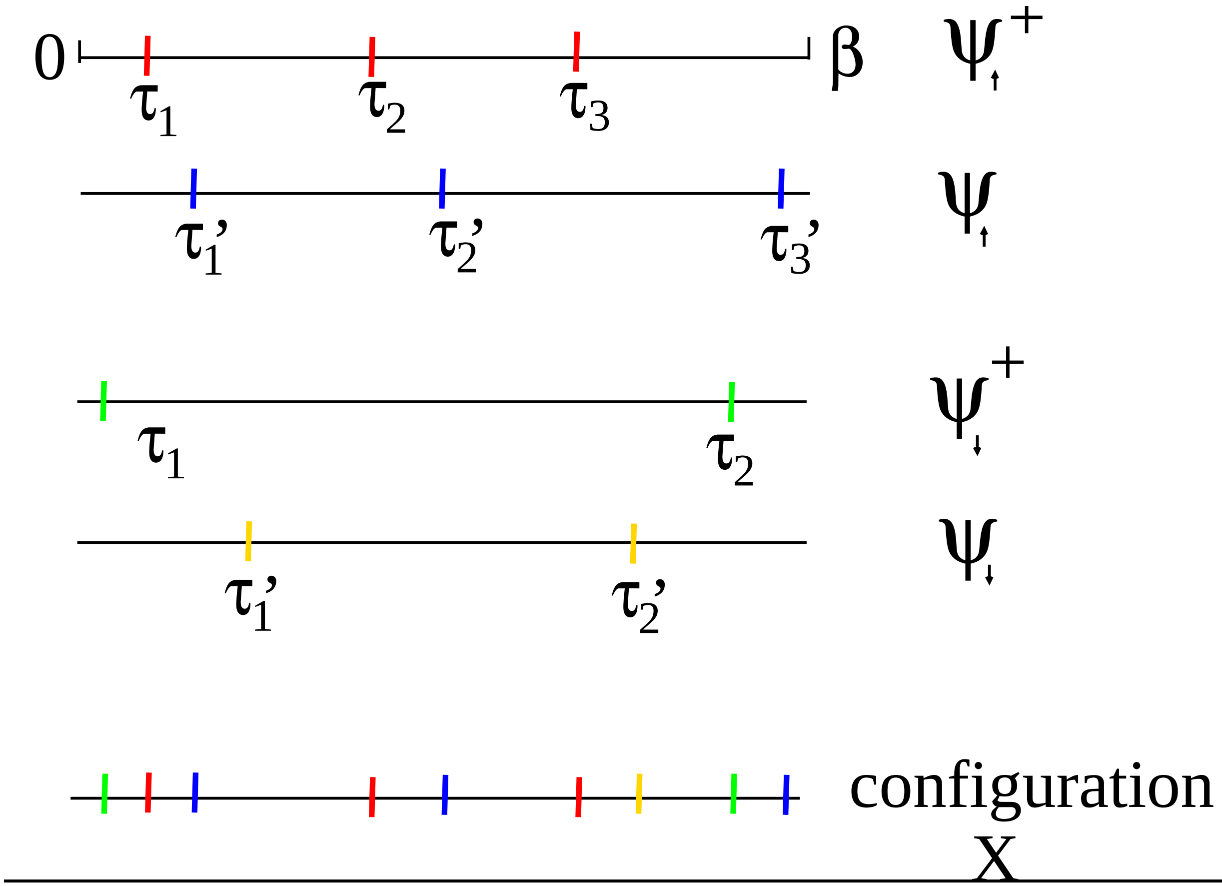
- Randomly chose perturbation order k
- Generate k random times in the interval $[0,\beta]$ for anhilation operator and k random times for construction operator in the same interval.
- Evaluate the weight (partition function) for this particular configuration of times (let's call it a diagram although it is a collection of diagrams)

$$Z_{\mathcal{D}} = Z_a \frac{1}{(k!)^2} \langle T_\tau \psi_{\alpha'_1}(\tau'_1)\psi^\dagger_{\alpha_1}(\tau_1)\cdots\psi_{\alpha'_k}(\tau'_k)\psi^\dagger_{\alpha_k}(\tau_k)\rangle_{local} \times$$
$$\times Det\begin{pmatrix} \Delta_{\alpha_1\alpha'_1}(\tau_1,\tau'_1) & \Delta_{\alpha_1\alpha'_2}(\tau_1,\tau'_2) & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ \Delta_{\alpha_k\alpha'_1}(\tau_k,\tau'_1) & \cdots & \cdots & \Delta_{\alpha_k\alpha'_k}(\tau_k,\tau'_k) \end{pmatrix}. \quad (65)$$

The trace in $\langle\cdots\rangle_{local}$ should be properly time-ordered. The move should be accepted with the probability proportional to its partition function contribution.

This type of sampling woule be extremely inefficient. Exponentially many diagrams must be visited with Monte Carlo importance sampling. To achieve that, a set of simple moves is implemented which can be made sufficiently fast so that the large phase space is sampled efficiently.

Configurations which are being sampled with Monte Carlo can be labeled by a set of times corresponding to various anhilation or creation operators



Each application of the operator ψ makes a ”kink” in the time evolution of the atom therefore we will call kink to be application of any operator in the time evolution.

The times on each line can be time ordered and only the time ordered combination needs to be visited. All the rest of the configurations are automatically included because of the determinant of bath Δ in the expression which takes into account all possible ways of connecting τ_i with τ'_j through bath $\Delta(\tau_i - \tau'_j)$.

The configurations $X = [[\tau_0, \tau_1, \cdots, \tau_k], [\tau'_0, \tau'_1, \cdots, \tau'_k], \cdots]$ can be visited with the set of simple moves. The following are usually used:

- Insertion of one creation and one destruction operator of the same flavor (spin up or spin down) at random times.
- Removal of two randomly choosen times τ_i and τ'_j where one corresponds to anhilation and one to creation operator of the same flavor (up or down).
- Shift of one operator in time (shift of a kink).
- Insertion of multiple kinks
- Removal of multiple kinks

The detailed balance condition requires that the probability to insert two kinks at random times τ, τ', being chosen uniformly in the interval $[0, \beta)$, is

$$P_{add} = min\left[\left(\frac{\beta\, N_b}{k+1}\right)^2 \frac{\mathcal{Z}_{new}}{\mathcal{Z}_{old}}\frac{\mathcal{D}_{new}}{\mathcal{D}_{old}}, 1\right] \tag{66}$$

where N_b is the number of baths, k is the current perturbation order (number of kinks/2), $\mathcal{Z}_{new}$ is the local matrix element

$$\begin{aligned}\mathcal{Z}_{new} = \langle T_\tau \psi_{\alpha'_{new}}(\tau'_{new})\psi^\dagger_{\alpha_{new}}(\tau_{new})\psi_{\alpha'_1}(\tau'_1)\psi^\dagger_{\alpha_1}(\tau_1)\cdots \\ \cdots\psi_{\alpha'_k}(\tau'_k)\psi^\dagger_{\alpha_k}(\tau_k)\rangle_{local}\end{aligned} \tag{67}$$

and $\mathcal{D}_{new}/\mathcal{D}_{old}$ is the ratio between the new and the old determinant of baths Δ. The factors of (βN_b) enter because of the increase of the phase space when adding kinks (increase of the entropy) Similarly, the probability to remove two kinks, chosen randomly between $[1\cdots k]$ is

$$P_{remove} = min\left[\left(\frac{k}{\beta\, N_b}\right)^2 \frac{\mathcal{Z}_{new}}{\mathcal{Z}_{old}}\frac{\mathcal{D}_{new}}{\mathcal{D}_{old}}, 1\right]. \tag{68}$$

One can explain these probabilities also as a product of trial-step probability $\omega_{X\to X'}$ and acceptance probability $A_{X\to X'}$. The detail balance condition requires

$$\frac{A_{X_{2k}\to X_{2k+2}}}{A_{X_{2k+2}\to X_{2k}}} = \frac{\rho(X_{2k+2})}{\rho(X_{2k})}\frac{\omega_{X_{2k+2}\to X_{2k}}}{\omega_{X_{2k}\to X_{2k+2}}} \tag{69}$$

When adding two kinks $X_{new}\equiv X_{2k+2}$ has $2k+2$ kinks while $X_{old}\equiv X_{2k}$ has $2k$ kinks. The trial step probability for a random insertion in the $[0,\beta]$ interval is

$$\omega_{X_{2k}\to X_{2k+2}} = \frac{d\tau_{new}d\tau'_{new}}{\beta^2}$$

while removal of the two kinks, where we need to select one creation operator among $k+1$ creation operators, and one anhilation operator among $k+1$ anhilation operators, is

$$\omega_{X_{2k+2}\to X_{2k}}\frac{1}{(k+1)^2}$$

The probability distribution $\rho(X)$ is proportional to X's contribution the partition function, and is

$$\rho(X_{2k}) = d\tau_1\cdots d\tau_k d\tau'_1\cdots d\tau'_k \mathcal{Z}_{2k}\mathcal{D}_{2k} \tag{70}$$

$$\rho(X_{2k+2}) = d\tau_1\cdots d\tau_k d\tau'_1\cdots d\tau'_k\, d\tau_{new}d\tau_{new'}\mathcal{Z}_{2k+2}\mathcal{D}_{2k+2} \tag{71}$$

(72)

The net acceptance probability is therefore

$$\frac{A_{X_{2k+2}\rightarrow X_{2k}}}{A_{X_{2k}\rightarrow X_{2k+2}}} = \frac{\beta^2 d\tau_{new} d\tau'_{new} \mathcal{Z}_{2k+2}\mathcal{D}_{2k+2}}{(k+1)^2 d\tau_{new} d\tau'_{new} \mathcal{Z}_{2k}\mathcal{D}_{2k}}$$

which enters the above stated probability to add two kinks P_{add}.

The only extra variable in the above equation is N_b, which is the dimensionality of the bath. For one-band model, N_b is unity. In case the spin is not a good quantum number (like superconductivity), than N_b would be 2 for the composite bath of $\uparrow$ and $\downarrow$ spins. In the latter case, we would need to consider the off-diagonal hybridizations and off-diagonal operators $\psi^\dagger_\uparrow \psi_\downarrow \Delta_{\uparrow\downarrow}$. In this case, we need to add one kink in the $[0, \beta]$ interval of the $\uparrow$-spins or $[0, \beta]$ interval of $\downarrow$ spins. Effectively, the interval is thus increased by factor of two (we have two choices). In general it increases by the dimensionality of the bath N_b and the interval is thus of size βN_b.

To efficiently sample over a large Hilbert space of diagrams, we need to learn how to compute the ratios $\frac{\mathcal{Z}_{new}}{\mathcal{Z}_{old}}$ and $\frac{\mathcal{D}_{new}}{\mathcal{D}_{old}}$ with the ”fast update” type of formulas.

3.5 The local part of the partition function

Exact Diagonalization of the Atom/Cluster

To efficiently evaluate the atomic part of the partition function $\mathcal{Z}_{new}$ one can diagonalize the atomic part of the Hamiltonian, i.e., $H_a = E_m|m\rangle$.

It is also crucial to take into account the conservation of various quantum numbers, such as the particle number, the total spin, and the total momentum of the atom/cluster states when evaluating the atomic traces.

Typical contribution to the atomic part of the trace that needs to be evaluated at each Monte Carlo step, takes the form

$$\mathcal{Z}_{\mathcal{D}} = \mathrm{Tr}\left(T_\tau e^{-\beta H_a}\psi_{\alpha_n}(\tau'_n)\psi^\dagger_{\alpha_{n-1}}(\tau_n)\cdots\psi_{\alpha_2}(\tau'_1)\psi^\dagger_{\alpha_1}(\tau_1)\right) \tag{73}$$

$$= \sum_{\{m\}} e^{-(\beta-\tau'_n)E_{m_1}}(\psi_{\alpha_n})_{m_1 m_2} e^{-(\tau'_n-\tau_n)E_{m_2}}(\psi^\dagger_{\alpha_{n-1}})_{m_2 m_3}\cdots$$

$$(\psi_{\alpha_2})_{m_{n-1}m_n} e^{-(\tau'_1-\tau_1)E_{m_n}}(\psi^\dagger_{\alpha_1})_{m_n m_1} e^{-\tau_1 E_{m_1}}$$

where the matrix elements are $(\psi^\dagger_{\alpha_i})_{nm} = \langle n|\psi^\dagger_{\alpha_i}|m\rangle$ and E_m are eigenvalues of the atom. The actual order of operators in Eq. (73) depends on their time arguments and creation operator is not necessary followed by annihilation operator. Note that we are using the Heisenberg representation for operators, i.e., $\psi_\alpha(\tau) = e^{\tau H_a}\psi_\alpha e^{-\tau H_a}$.

This approach is extremely efficient for a single orbital and a few orbital model. In simple one band model (or whenever ψ operator connects one atomic eigenstates with only one atomic eigenstate), we need to multiply numbers only, and the entire product is either zero, or just the exponent of the sum $\pm\exp(\sum_i -\Delta\tau_i E_{n_i})$, where $\Delta\tau$ is the time interval between kinks, and E_n is the atomic eigenvalue in the chain product.

But when the complexity of the atom grows, the bottleneck becomes the fast growth in the number of atomic states $|m\rangle$ (for example, single site DMFT for the f shell requires 2^{14} states). Consequently, the matrix elements ψ are in general very large matrices and the typical diagram order is inversely proportional to temperature therefore one typically needs to multiply few hundred large matrices at each Monte Carlo step. It is clear that this is very impractical and we need tricks like:

- Most of matrix elements vanish due to Pauli principle. A fast algorithm is needed to determine which matrix elements are nonzero.
- Symmetries of the problem can be taken into account to reduce the size of the ψ matrix.
- The number of trial steps is usually much bigger (100 times) than the number of

accepted steps and the insertion or removal of a kink is very local-in-time operation. It is convenient to store some of the product in Eq. 73 (in a priority queue or tree structure, or just as linear chains) and when trying to insert a new kink, recompute only $\log(n)$ products.

Concept of Superstates:

The eigenstates of the atom can be written in a form $|N, S_z; \gamma\rangle$, where N is total number of electrons in the state (one band model has only 0,1,2), S_z is the z component of the total spin and γ stands for the rest of the quantum numbers.

The creation operator satisfies conservation laws, for example $\psi^\dagger_\sigma|N, S_z\rangle = |N+1, S_z+\sigma\rangle$ because the creation operator is nonzero only between Hilbert subspace of $\{N, S_z\}$ and $\{N+1, S_z+\sigma\}$.

In more general type of impurity problem one can have more quantum numbers and more conservation laws: For example momentum can be a good quantum number and total spin,... These conservation laws can greatly reduce the size of matrix $(\psi^\alpha)_{nm}$. In the one band model, it becomes a number and in 4-site impurity model, the largest matrix is it 12×6 after conservation laws are taken into account.

It is convenient to group together states with the same conserved quantum numbers

$\{N, S_z\}$ and treat the rest of the quantum numbers as internal degrees of freedom of a atomic superstate $|i\rangle \equiv |\{N, S_z\}\rangle$. The superstate $|i\rangle$ can be multidimensional state with internal quantum numbers $|m[i]\rangle$. It is then clear that creation operator acting on a state $|i\rangle$ gives a unique state $|j\rangle = \psi_\sigma^\dagger |i\rangle$ and it is enough to store a single index array $F^{\alpha\dagger}(i) = j$ to figure out how the Hilbert subspaces are visited under application of a sequence of creation and annihilation operators such as in Eq. (73): $i_1 \rightarrow F^{\alpha_1 \dagger}(i_1) \rightarrow \cdots i_k = F^{\alpha_k}(i_{k-1})$. This sequence is very often truncated in few steps only, due to the Pauli principle, i.e., because most of the sequences contain either multiple application of the same creation or annihilation operator (Pauli principle) or because they lead to a state outside the base (for example $\psi|N = 0...\rangle = 0$ or $\psi^\dagger|N = max...\rangle = 0$).

Once the nonzero matrix elements are found by this simple index lookup, the value of the matrix element needs to be computed by matrix multiplication.

In general impurity problem, to compute the trace in Eq. (73) we start with unity matrix in each subspace of a superstate $|i\rangle$ and apply both the time evolution operator $e^{-E_m(\tau_l - \tau_l')}$ (by multiplying each row of a matrix with its time evolution) and the kink (by multiplication with the matrix $(\psi_\alpha)_{mn}$ or $(F_\alpha^\dagger)_{mn}$). The operation of ψ brings us to the next superstate

$F^\alpha(i)$ where we repeat both the time evolution and the kink application. After k steps, the trace of a matrix gives the desired matrix element.

Storing the Time Evolution:

The number of kinks is proportional to inverse temperature β and kinetic energy of the system $\langle k\rangle = \beta|E_{kin}|$ (see Eq. (133)). It thus becomes large at low temperatures. However, an insertion of a kink with large time difference ($\psi^\dagger(\tau)\psi(\tau')$ with large $\tau - \tau'$) has a very low probability. The reason is that Pauli principle forbids insertion of the pair $\psi^\dagger_\alpha(\tau)\psi_\alpha(\tau')$ if another kink of the same species α is between the two times (τ, τ'). At the same time, $\Delta(\tau)$ is like $G(\tau)$ peaked at small times $\tau - \tau'$ and falls of at large times making the long time intervals rare.

The insertion of a kink is thus fairly local in time operation, therefore it is convenient to store a whole chain of products that appear in Eq. (73) from both sides, left and right, to make trial step very cheap. It takes only few matrix multiplications (almost independent of temperature) to compute the trace in Eq. (73). When the move is accepted, the trace needs to be updated which takes somewhat more time. However, it turns out that the time evolution far from the interval being inserted/removed, is typically changed in a trivial way by just multiplication by a phase factor. Hence expensive multiplication far from the interval being inserted is very rarely done in practice.

More sophisticated techniques were implemented for storing these products (see for example PRB B 90, 075149 (2014)) . One is the implementation of the *skip list*, or a *binary tree*.

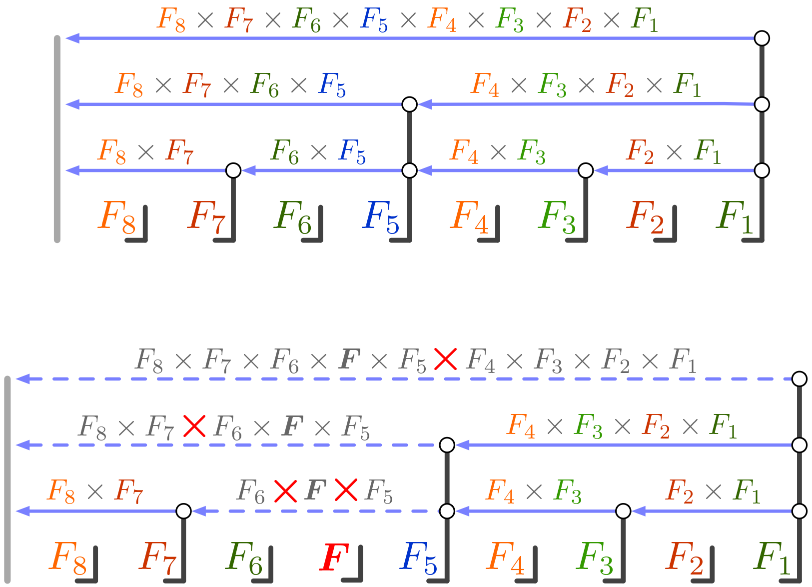


FIG. 2. (Color online) Top panel: Storage scheme for subproducts of matrices. The arrows store the products of matrices they span over. The $l = 1$ level stores the pair products, the $l = 2$ their products, and so on. Lower panel: The matrix F has been inserted in the matrix product of the top panel and the products with a bold red multiplication sign need to be calculated in order to obtain the total product.

3.6 The bath part of the partition function

The bath part of Z, denoted by $\mathcal{D}$ above, is

$$\mathcal{D} = Det\begin{pmatrix} \Delta_{\alpha_1\alpha'_1}(\tau_1,\tau'_1) & \Delta_{\alpha_1\alpha'_2}(\tau_1,\tau'_2) & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ \Delta_{\alpha_k\alpha'_1}(\tau_k,\tau'_1) & \cdots & \cdots & \Delta_{\alpha_k\alpha'_k}(\tau_k,\tau'_k) \end{pmatrix} \tag{74}$$

When adding (removing) two kinks, one column and one row is added to (removed from) $\mathcal{D}$. It is numerically unstable to compute determinant of a large matrix. For the transition probability, we need only the ratio $\mathcal{D}_{new}/\mathcal{D}_{old}$. This ratio is simpler and numerically more stable to compute.

It turns out that more useful than matrix of Δ's written above is its inverse

$$M = \begin{pmatrix} \Delta_{\alpha_1\alpha'_1}(\tau_1,\tau'_1) & \Delta_{\alpha_1\alpha'_2}(\tau_1,\tau'_2) & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ \Delta_{\alpha_k\alpha'_1}(\tau_k,\tau'_1) & \cdots & \cdots & \Delta_{\alpha_k\alpha'_k}(\tau_k,\tau'_k) \end{pmatrix}^{-1} \tag{75}$$

We will see later that this matrix is directly connected to the local Green's function. The ratio of determinants can be equally simply expressed by this matrix $\mathcal{D}_{new}/\mathcal{D}_{old} = Det(M_{old})/Det(M_{new})$.

In computer, we will store and manipulate only matrix M.

Shermann Morrison formulas:

This formula is used for fast inversion of a matrix. If we have matrix a A and we change A^{-1} with adding $u \otimes v$, the inversion of the results can be done in N^2 steps rather than usual N^3. The formula is

$$(A^{-1} + u \otimes v)^{-1} = A - \frac{1}{1 + vAu}(Au) \otimes (vA) \tag{76}$$

Here u and v are vectors.

Further, determinants can be expressed by

$$Det(A^{-1} + u \otimes v)Det(A) = 1 + vAu \tag{77}$$

3.6.1 Fast update for addition of a column and a row

We have a matrix M which is $(n-1)\times(n-1)$ matrix. We change the matrix in the following way

$$M_{new}^{-1} = \left(\begin{array}{ccc|c} & & & \Delta_{1n} \\ & M^{-1} & & \Delta_{2n} \\ & & & \cdots \\ \hline \Delta_{n1} & \Delta_{n2} & \cdots & \Delta_{nn} \end{array}\right) \tag{78}$$

Provided we know M, M_{new} can be obtained using the Shermann Morrison formulas.

We will derive the equation in two steps. First we will define an intermediate matrix $\widetilde{M}$, which is given by

$$\widetilde{M}^{-1} = \left(\begin{array}{ccc|c} & M^{-1} & & 0 \\ & & & 0 \\ & & & \cdots \\ \hline 0 & 0 & \cdots & 1 \end{array}\right) + \left(\begin{array}{c} \Delta_{1n} \\ \Delta_{2n} \\ \cdots \\ \hline \Delta_{nn} - 1 \end{array}\right) \otimes (0, 0, \cdots | 1) = \tag{79}$$

$$= \left(\begin{array}{ccc|c} & M^{-1} & & \Delta_{1n} \\ & & & \Delta_{2n} \\ & & & \cdots \\ \hline 0 & 0 & \cdots & \Delta_{nn} \end{array}\right) \tag{80}$$

Second, M_{new} is given by

$$M_{new}^{-1} = \widetilde{M}^{-1} + \left(\begin{array}{c} 0 \\ 0 \\ \cdots \\ \hline 1 \end{array}\right) \otimes (\Delta_{n1}, \Delta_{n2}, \cdots | 0) = \left(\begin{array}{ccc|c} & & & \Delta_{1n} \\ & M^{-1} & & \Delta_{2n} \\ & & & \cdots \\ \hline \Delta_{n1} & \Delta_{n2} & \cdots & \Delta_{nn} \end{array}\right) \tag{81}$$

In the first step we have

$$u = (\Delta_{1n}, \Delta_{2n}, \cdots | \Delta_{nn} - 1) \tag{82}$$

$$v = (0, 0, \cdots | 1) \tag{83}$$

$$Mu = (L_{1n}, L_{2n}, \cdots, \Delta_{nn} - 1) \tag{84}$$

$$vM = (0, 0, \cdots | 1) \tag{85}$$

$$vMu = \Delta_{nn} - 1 \tag{86}$$

where

$$L_{ij} = \sum_{l<n} M_{il} \Delta_{lj} \tag{87}$$

Using the above formula, it is easy to see that

$$\widetilde{M} = M - \frac{1}{\Delta_{nn}} \left(\begin{array}{c} L_{1n} \\ L_{2n} \\ \cdots \\ \hline \Delta_{nn} - 1 \end{array} \right) \otimes (0, 0, \cdots | 1) = \left(\begin{array}{ccc|c} & M & & -L_{1n}/\Delta_{nn} \\ & & & -L_{2n}/\Delta_{nn} \\ & & & \cdots \\ \hline 0 & 0 & \cdots & 1/\Delta_{nn} \end{array} \right) \quad (88$$

In the second step we have

$$u = (0, 0, \cdots | 1) \quad (89)$$

$$v = (\Delta_{n1}, \Delta_{n2}, \cdots | 0) \quad (90)$$

$$\widetilde{M} u = (-L_{1n}/\Delta_{nn}, -L_{2n}/\Delta_{nn}, \cdots, 1/\Delta_{nn}) \quad (91)$$

$$v \widetilde{M} = (R_{n1}, R_{n2}, \cdots | -q/\Delta_{nn}) \quad (92)$$

$$v M u = -q/\Delta_{nn} \quad (93)$$

where

$$L_{ij} = \sum_{l<n} M_{il} \Delta_{lj} \quad (94)$$

$$
\begin{aligned}
R_{ij} &= \sum_{l<n} \Delta_{il}\, M_{lj} && (95)\\
q &= \sum_{l<n} \Delta_{nl} L_{ln} = \sum_{ll'<n} \Delta_{nl}\, M_{ll'}\, \Delta_{l'n} && (96)
\end{aligned}
$$

Finally,

$$
M_{new} = \widetilde{M} - \frac{1}{1-q/\Delta_{nn}} \left(\begin{array}{c} -L_{1n}/\Delta_{nn} \\ -L_{2n}/\Delta_{nn} \\ \cdots \\ \hline 1/\Delta_{nn} \end{array} \right) \otimes \left(R_{n1}, R_{n2}, \cdots | -q/\Delta_{nn}\right) \quad (97)
$$

Combining the two equations for $\widetilde{M}$ and M_{new} we have

$$
M_{new} = \left(\begin{array}{ccc|c} & & & 0 \\ & M & & 0 \\ & & & \cdots \\ \hline 0 & 0 & \cdots & 0 \end{array} \right) + \left(\begin{array}{ccc|c} & & & -p\, L_{1n} \\ & \Lambda & & -p\, L_{2n} \\ & & & \cdots \\ \hline -p\, R_{n1} & -p\, R_{n2} & \cdots & p \end{array} \right) \quad (98)
$$

where matrix Λ and p are

$$\frac{1}{p} = \Delta_{nn} - q = \Delta_{nn} - \sum_{l<n} \Delta_{nl} L_{ln} = \Delta_{nn} - \sum_{ll'<n} \Delta_{nl}\, M_{ll'}\, \Delta_{l'n} \quad (99)$$

$$\Lambda_{ij} \equiv p\, L_{in} R_{nj} \quad (100)$$

It is more efficient to express M_{new} by

$$M_{new} = \left(\begin{array}{ccc|c} & & & 0 \\ & M & & 0 \\ & & & \cdots \\ \hline 0 & 0 & \cdots & 0 \end{array}\right) + p \left(\begin{array}{c} L_{1n} \\ L_{2n} \\ \cdots \\ \hline -1 \end{array}\right) \otimes (R_{n1}, R_{n2}, \cdots | -1) \quad (101)$$

This is again the ”rank 1” update formula and can be implemented by BLAS routine dger.

Finally, the ratio of determinants is

$$Det(\widetilde{M}^{-1}) Det(M) = \Delta_{nn} \quad (102)$$

$$Det(M_{new}^{-1}) Det(\widetilde{M}) = \frac{1}{p\, \Delta_{nn}} \quad (103)$$

therefore

$$\frac{\mathcal{D}_{new}}{\mathcal{D}_{old}} = \frac{Det(M)}{Det(M_{new})} = \frac{1}{p} = \Delta_{nn} - \sum_{ll'<n} \Delta_{nl}\, M_{ll'}\, \Delta_{l'n} \tag{104}$$

3.6.2 Removal of a row and a column

Original matrix M is $n \times n$ and we want to remove the coulmn and row number k in the inverse of this matrix.

The formulas can again be derived by Shermann-Morrison formulas or can be checked by mathematica

$$M_{ij}^{new} = M_{ij} - \frac{M_{ik}M_{kj}}{M_{kk}} \tag{105}$$

where i and j now do not need to run over index k. The ratio of determinants is

$$\frac{\mathcal{D}_{new}}{\mathcal{D}_{old}} = \frac{Det(M)}{Det(M_{new})} = M_{kk} \tag{106}$$

In more general case, if we remove a column i and a row j from Δ, the ratio of determinants is

$$\frac{\mathcal{D}_{new}}{\mathcal{D}_{old}} = (-1)^{chessboard} M_{ij} \tag{107}$$

where $(-1)^{chessboard}$ is unity when the corresponding field (i, j) on the chessboard is dark, and is (-1) when this field is white.

3.7 Green's function Evaluation:

The local Green's function can be computed in two ways:

- Directly from the local trace, i.e.,

$$G(\tau_1-\tau_2)=\frac{1}{Z}\mathrm{Tr}(T_\tau e^{-\int_0^\beta H d\tau}\psi_\sigma(\tau_1)\psi^\dagger_\sigma(\tau_2))$$

  which can be evaluated just like to local trace above with the only difference of two more kinks in the interval $[0,\beta]$. This requires *Worm algorithm* (two dangling ends i.e., Ira and Masha, correspond to ψ and $\psi^\dagger$). This will not be further discussed here.

- Through the conduction electron green's function using the same set of kinks as needed for computing Z.

It is much more efficient to compute the electron Green's function by the second method. We first recognize that the impurity Green's function can be computed by the logarithmic derivative of Z (Eq. 30) with respect to the hybridization Δ, i.e.,

$$G_{\beta'\beta}(\tau_i'-\tau_j)=-\frac{\delta\log Z}{\delta\Delta_{\beta\beta'}(\tau_j-\tau_i')}= \tag{108}$$

$$\frac{1}{Z}\int D[\psi^\dagger\psi]e^{-S_a-\int_0^\beta d\tau\int_0^\beta d\tau'\sum_{\alpha\alpha'}\psi^\dagger_\alpha(\tau)\Delta_{\alpha\alpha'}(\tau-\tau')\psi_{\alpha'}(\tau')}\psi^\dagger_\beta(\tau_j)\psi_{\beta'}(\tau'_i)$$

where S_a is the atomic action.

The derivative of the Eq. (60), which is sampled by the CTQMC, is

$$\frac{\delta\log(Z)}{\delta\Delta_{\beta\beta'}(\tau_j-\tau'_i)} = \frac{1}{Z}Z_a\sum_k\frac{1}{k!}\sum_{\alpha_1\cdots\alpha'_k}\prod_{l=0}^{k}\int_0^\beta d\tau_l\int_0^\beta d\tau'_l$$
$$\times\langle T_\tau\psi_{\alpha'_1}(\tau'_1)\psi^\dagger_{\alpha_1}(\tau_1)\cdots\psi_{\alpha'_k}(\tau'_k)\psi^\dagger_{\alpha_k}(\tau_k)\rangle_{local}\times$$
$$\times\frac{1}{k!}\frac{\delta}{\delta\Delta_{\beta\beta'}(\tau_j-\tau'_i)}Det\begin{pmatrix}\Delta_{\alpha_1\alpha'_1}(\tau_1,\tau'_1) & \cdots & \cdots\\ \cdots & \cdots & \cdots\\ \cdots & \cdots & \cdots\\ \Delta_{\alpha_k\alpha'_1}(\tau_k,\tau'_1) & \cdots & \Delta_{\alpha_k\alpha'_k}(\tau_k,\tau'_k)\end{pmatrix} \quad (109)$$

Next we recognize that the derivative of the determinant gives nonzero contribution when $\Delta_{\beta\beta'}(\tau_j-\tau'_i)$ appears in the determinant. In this case, the results is a similar determinant, but with one column and one row removed from the determinant. The removed row and the removed column both contain $\Delta_{\beta\beta'}(\tau_j-\tau'_i)$. In addition, there is a minus sign of a chessboard pattern when $\Delta_{\beta\beta'}(\tau_j - \tau_i')$ is on the white field.

Let us use the notation

$$\underline{\Delta} \equiv M^{-1} = \begin{pmatrix} \Delta(\tau_1, {\tau'}_1) & \cdots \\ \cdots & \cdots \\ \cdots & \Delta(\tau_k, {\tau'}_k) \end{pmatrix} \tag{110}$$

and write the expression being sampled by CTQMC

$$\frac{\delta \log(Z)}{\delta\Delta(\tau_j - \tau_i')} = \frac{1}{Z} Z_a \sum_k \frac{1}{k!} \sum_{\alpha_1 \cdots \alpha_k'} \prod_{l=0}^{k} \int_0^\beta d\tau_l \int_0^\beta d\tau_l'$$

$$\times \langle T_\tau \psi_{\alpha_1'}(\tau_1')\psi^\dagger_{\alpha_1}(\tau_1) \cdots \psi_{\alpha_k'}(\tau_k')\psi^\dagger_{\alpha_k}(\tau_k)\rangle_{local} \frac{1}{k!} Det(\underline{\Delta}) \times$$

$$\times(-1)^{chessboard}\frac{Det\left(\begin{array}{cc|c|cc}
\Delta(\tau_1,\tau_1') & \cdots & \cancel{\Delta(\tau_1,\tau_i')} & \cdots & \Delta(\tau_1,\tau_k') \\
\cdots & \cdots & \cdots & \cdots & \\
\hline
\cancel{\Delta(\tau_j,\tau_1')} & \cdots & \cancel{\Delta(\tau_j-\tau_i')} & \cdots & \cancel{\Delta(\tau_j,\tau_k')} \\
\hline
\cdots & \cdots & \cdots & \cdots & \\
\Delta(\tau_k,\tau_1') & \cdots & \cancel{\Delta(\tau_k-\tau_i')} & \cdots & \Delta(\tau_k,\tau_k')
\end{array}\right)}{Det\,(\underline{\Delta})}\tag{111}$$

We showed above that the removal of a column and a row gives the matrix element of the inverse $\underline{\Delta}^{-1} \equiv M$ at the center of the cross, i.e., $M_{\tau_i',\tau_j}(-1)^{chessboard}$.

Next we recognice that all terms but the ratio of determinats appear also in computing Z, hence they are part of the Monte Carlo weight W_{MC}. Hence during Monte Carlo sampling we just need to sample the matrix element M_{τ_i',τ_j}, i.e.,

$$G(\tau_i'-\tau_j) = -\sum_{MC} W_{MC} M_{\tau_i',\tau_j}\tag{112}$$

We can Fourier transform each contribution to the Green's function to obtain

$$G(i\omega) = -\sum_{MC} W_{MC} \frac{1}{\beta} \sum_{i,j} e^{i\omega\tau_i'} M_{\tau_i',\tau_j} e^{-i\omega\tau_j}. \tag{113}$$

We can thus sample Green's function directly in imaginary frequency, rather then imaginary time. This equation is the central equation of the approach since it relates the local Green's function with the quantities computed in QMC importance sampling.

The equation (113) shows that only matrix $M \equiv (\underline{\Delta})^{-1}$ needs to be stored and manipulated in the simulation.

To derive the formula for the impurity green's function, we used the first order derivative of Z with respect to the hybridization. If one perform the second order derivatives with respect to the hybridization Δ, we obtain formulas for the two particle susceptibilities. These are important quantities that eneter many two particle physical observables, and are readily available in the CTQMC approach.

Fast Updates: The Green's function can be updated in linear time (rather than quadratic). When adding a construction and annihilation operator at τ and τ', adding a column at i_s and row at i_e to matrix M^{-1}, leads to the following update formula for the local Green's function

$$G^{new} = G^{old} - \frac{p}{\beta}\left(\sum_{j_e} e^{i\omega\tau_{j_e}} L_{j_e}\right)\left(\sum_{j_s} R_{j_s} e^{-i\omega\tau_{j_s}}\right). \tag{114}$$

Quantities L and R were defined above. It is clear from Eq. (114) that only linear amount of time is needed to update the local Green's function.

When removing two kinks of construction and annihilation operators at i_s and i_e, Green's functions are related by

$$G^{new} = G^{old} + \frac{1}{\beta M_{i_e i_s}}\left(\sum_{j_e} e^{i\omega\tau_{j_e}} M_{j_e i_s}\right)\left(\sum_{j_s} M_{i_e j_s} e^{-i\omega\tau_{j_s}}\right). \tag{115}$$

Finally, the exponential factors $e^{i\omega\tau_i}$ do not need to be recomputed at each Monte Carlo step since all ”old” times can be stored and the exponents need to be computed only for the new pair of times and only at each accepted move.

It is important that one can sample directly $G(i\omega)$ rather than $G(\tau)$ to avoid discretization error due to binning of $G(\tau)$.

3.7.1 High Frequencies and Moments:

Similarly to the Hirsch-Fye QMC, the low frequency points of Green’s function converge very fast to the exact value while the high-frequency points, when sampled directly, contain a lot of noise. It is therefore not very useful to sample large frequencies in the above described way. Usually we sample low frequency Matsubara points, while the high-frequency are replaced by the high frequency moments of the self-energy computed analytically. What is low frequency? A good rule of a tumb is that $\omega_n^{max} L \approx 10$, where L is the high frequency cutoff on the real axis. This results in $(2n^{max}+1)(\pi/\beta)L \approx 10$ or $n^{max} \approx 3(\beta/L)$.

The high frequency moments of the self-energy are computed from the Green’s function moments, which, in general, take the following form

$$m_n^{\alpha\beta} = (-1)^n \left\langle \left\{ [H,[H,\cdots[H,\psi_\alpha]\cdots]],\psi_\beta^\dagger \right\} \right\rangle \tag{116}$$

To compute the moments within the present approach, few operators need to be sampled in simulation. In the one-band model, only density is required, but in more complicated situation, higher order density-density and exchange terms enter. In general, Green's function moments can be expressed by the average of a few equal time operators.

It turns out that up to the second order in self-energy, the hybridization cancels out of the expression, and therefore H in the above equation can be replaced by H_a, which gives extremely simple form for this moments in the local eigenbasis

$$m_{1,mm'}^{G,\alpha\beta} = \sum_n (F^\alpha)_{mn}(F^{\beta\dagger})_{nm'}(E_n - E_m) \tag{117}$$
$$-(F^{\beta\dagger})_{mn}(F^\alpha)_{nm'}(E_n - E_{m'})$$
$$m_{2,mm'}^{G,\alpha\beta} = \sum_n (F^\alpha)_{mn}(F^{\beta\dagger})_{nm'}(E_m - E_n)^2 \tag{118}$$
$$+(F^{\beta\dagger})_{mn}(F^\alpha)_{nm'}(E_n - E_{m'})^2.$$

The lowest order self-energy moments can then be computed in the following way

$$\Sigma_{\alpha\beta}(\infty) = \langle m_1^{G,\alpha\beta} \rangle - E_{imp} \tag{119}$$

$$\Sigma^{(1)}_{\alpha\beta} = \langle m_2^{G,\alpha\beta} \rangle - \langle m_1^{G,\alpha\beta} \rangle^2 . \tag{120}$$

3.8 Sampling of zero time quantities

Equal time operators like those in Eq. (117) and (118) can be straightforwardly computed in simulation. Any cluster operator can be expressed in terms of Hubbard operators $X_{mm'}$, which project to the cluster eigenbase, and matrix elements of the operator in the cluster eigenbase

$$O = \langle m|O|m'\rangle X_{mm'} . \tag{121}$$

To compute the expectation value of the operator, the following probabilities are needed

$$P_{mm'} \equiv \langle X_{mm'} \rangle . \tag{122}$$

The expectation value of any cluster operator is then given by

$$\langle O \rangle = \sum_{mm'} \langle m|O|m'\rangle P_{mm'} . \tag{123}$$

To improve the sampling precision, operators are averaged over all times, i.e.,

$$P_{mm'} = \frac{1}{\beta}\int_0^\beta \langle X_{mm'}(\tau_0)\rangle d\tau_0 \qquad (124)$$

$$= \frac{1}{Z}\frac{1}{\beta}\int_0^\beta d\tau_0 \mathrm{Tr}[T_\tau e^{-\int_0^{\tau_0} H d\tau} X_{mm'}(\tau_0) e^{-\int_{\tau_0}^{\beta} H d\tau}]$$

If the time τ_0 is between two kinks $\tau_l < \tau_0 < \tau_{l+1}$ we have

$$P_{mm'} = \frac{1}{Z\beta}\sum_l \int_{\tau_l}^{\tau_{l+1}} d\tau \mathrm{Tr}[T_\tau e^{-\int_0^{\tau_l} H d\tau} e^{-H_a(\tau-\tau_l)} X_{mm'} e^{-H_a(\tau_{l+1}-\tau)} e^{-\int_{\tau_{l+1}}^{\beta} H$$

$$= \frac{1}{Z\beta}\sum_l \sum_{m_0} \langle m_0|e^{-\int_0^{\tau_l} H d\tau}|m\rangle \int_{\tau_l}^{\tau_{l+1}} d\tau e^{-E_m(\tau-\tau_l)} e^{-E_{m'}(\tau_{l+1}-\tau)} \langle m'|e^{-\int_{\tau_{l+1}}^{\beta} H d\tau}|$$

$$= \frac{1}{Z}\sum_l \sum_{m_0} \langle m_0|e^{-\int_0^{\tau_l} H d\tau}|m\rangle w_{mm'}(\tau_{l+1}-\tau_l) \langle m'|e^{-\int_{\tau_{l+1}}^{\beta} H d\tau}|$$

Here

$$w_{mm'}(\Delta\tau) = \frac{1}{\beta}\frac{e^{-E_m\Delta\tau} - e^{-E_{m'}\Delta\tau}}{E_{m'} - E_m} \qquad (127)$$

To improve the speed in computation of local traces, the quantities

$$\langle m_0|e^{-\int_0^{\tau_l} H d\tau}|m\rangle$$

are stored and regularly updated during simulation.

Further simplification is possible if operator $\langle O\rangle$ commutes with H_a because only the diagonal probabilities P_{mm} are needed in this case. For example, the electron density n_f and average magnetization $\langle S_z\rangle$ are two important quantities which can be computed to very high precision in this way. For example

$$n_f = \sum_m P_m \langle m|\hat{N}|m\rangle \tag{128}$$

$$M = \sum_m P_m \langle m|\hat{S}^z|m\rangle \tag{129}$$

Here diagonal probabilities are

$$P_m = \sum_l \frac{\Delta\tau_l}{\beta}\frac{1}{Z}\sum_{m_0}\langle m_0|e^{-\int_0^{\tau_l} H d\tau}|m\rangle e^{-E_m\Delta\tau_l}\langle m|e^{-\int_{\tau_{l+1}}^{\beta} H d\tau}|m_0\rangle$$

and are just the weighted sum of the local trace contributions we derived above to compute the transition probabilities. The local trace in this notation is

$$\mathcal{Z} = \frac{1}{Z}\sum_{m_0,l}\langle m_0|e^{-\int_0^{\tau_l} H d\tau}|m\rangle e^{-E_m\Delta\tau_l}\langle m|e^{-\int_{\tau_{l+1}}^{\beta} H d\tau}|m_0\rangle$$

and becomes clear that the probability for each atomic state is proportional to the time each atomic state ”survives” in diagram.

3.9 Sampling the total energy:

The average of the potential energy $\langle V\rangle$ can be computed from the average of the local energy since

$$\begin{aligned}\langle H_{loc}\rangle &= \sum_{\alpha\beta\gamma\delta} U_{\alpha\beta\gamma\delta}\langle\psi^\dagger_\alpha\psi^\dagger_\beta\psi_\delta\psi_\gamma\rangle + \sum_{\alpha\beta} E_{imp\,\alpha\beta}\langle\psi^\dagger_\alpha\psi_\beta\rangle \\ &= \langle V\rangle + \mathrm{Tr}[E_{imp}n] \end{aligned} \tag{130}$$

Kinetic energy of the general single site DMFT $E_{kin} = \mathrm{Tr}[H^0_{\mathbf{k}}G_{\mathbf{k}}]$ can be computed by

$$E_{kin} = \mathrm{Tr}[\Delta G] + \mathrm{Tr}[(\mu + E_{imp})n] \tag{131}$$

The total energy is therefore given by

$$\langle H\rangle = \langle H_{loc}\rangle + \mathrm{Tr}[\Delta G] + \mu n \tag{132}$$

The first term H_{loc} can be computed very precisely in simulation. The sampled quantity $\langle O\rangle$ is just the energy of an atomic state and can be simply obtained from the probabilities for atomic states $\langle H_{loc}\rangle = \sum_{m\in all-states} P_m E_m$. Computing kinetic energy from the Green's function gives in general worse accuracy because the high-frequency behavior of the Green's function can not be directly sampled and augmentation with analytically computed tails is necessary. However, it is simple to show that the average value of the perturbation order is related to the average of the kinetic energy

$$\langle k \rangle = -\frac{1}{T}\mathrm{Tr}[\Delta G] \tag{133}$$

where $\langle k \rangle$ is the average perturbation order and T is temperature. The later quantity is directly sampled in the present algorithm and it is just the center of gravity of the histogram.

To show that we write expression for the average perturbation order

$$\langle k \rangle = \frac{1}{Z}\int D[\psi^\dagger\psi]e^{-S_a}\sum_k k\frac{(-1)^k}{k!}(\Delta S)^k \tag{134}$$

$$\langle k \rangle = -\frac{1}{Z}\int D[\psi^\dagger\psi]e^{-S_a}\sum_k \frac{(-1)^{k-1}}{(k-1)!}(\Delta S)^{k-1}\Delta S = -\langle \Delta S \rangle \tag{135}$$

which is equivalent to Eq. (133)

Finally, the total energy E is given by

$$E_{tot} = \langle H_{local} \rangle - T\langle k \rangle + \mu n \tag{136}$$

All quantities in this equation can be computed to very high accuracy and since low temperatures can be reached in this method, the entropy can be obtained by integrating the specific heat.

3.10 Sampling of other Response Functions

A general one or two particle response function can be written in the form

$$\chi(\tau_1,\tau_2) = \langle T_\tau A(\tau_1) B(\tau_2)\rangle \tag{137}$$

This includes one particle green's function and various type of susceptibilities.

We will start by rewriting the equation in the form of time evolution operator

$$\chi(\tau_1,\tau_2) = \frac{1}{Z}\mathrm{Tr}\left(e^{-\int_0^{\tau_1} H(\tau)d\tau} A(\tau_1) e^{-\int_{\tau_1}^{\tau_2} H(\tau)d\tau} B(\tau_2) e^{-\int_{\tau_2}^{\beta} H(\tau)d\tau}\right) \tag{138}$$

The time evolution $e^{-\int_0^\beta H(\tau)d\tau}$ includes kinks due to hybridization which were addressed above when sampling the partition function. In each diagram, we have $2k$ kinks and when computing response functions, we need to add two more operators at any combination of times τ_1, τ_2.

If the time lies between two kinks, lets say $\tau_{l_1} < \tau_1 < \tau_{l_1+1}$ and $\tau_{l_2} < \tau_2 < \tau_{l_2+1}$ we might rewrite

$$\chi(i\omega) = \frac{1}{Z\beta}\sum_{l_1,l_2}\int_{\tau_{l_1}}^{\tau_{l_1+1}} d\tau_1 \int_{\tau_{l_2}}^{\tau_{l_2+1}} d\tau_2 e^{i\omega(\tau_1-\tau_2)} \times \tag{139}$$

$$\mathrm{Tr}\left(e^{-\int_0^{\tau_{l_1}} H(\tau)d\tau} e^{-H_a(\tau_1-\tau_{l_1})} A(\tau_1) e^{-H_a(\tau_{l_1+1}-\tau_1)} e^{-\int_{\tau_{l_1+1}}^{\tau_{l_2}} H(\tau)d\tau}\right. \tag{140}$$

$$\left. e^{-H_a(\tau_2-\tau_{l_2})} B(\tau_2) e^{-H_a(\tau_{l_2+1}-\tau_2)} e^{-\int_{\tau_{l_2}}^{\beta} H(\tau)d\tau}\right) \tag{141}$$

because we know there is no kink between τ_{l_i} and τ_{l_i+1}. Here H_a is atomic part of H which can be exactly diagonalized $H_a|m\rangle = E_m|m\rangle$.

We can express any local operator in terms of the matrix elements of the atomic states, i.e.,

$$A = \sum_{mm'} |m\rangle A_{mm'} \langle m'| \tag{142}$$

Iserting the expansion of A and B we get

$$\chi(i\omega) = \frac{1}{Z\beta}\sum_{l_1,l_2}\sum_{\{m\}}\int_{\tau_{l_1}}^{\tau_{l_1+1}} d\tau_1 \int_{\tau_{l_2}}^{\tau_{l_2+1}} d\tau_2 e^{\tau_1(i\omega - E_m + E_{m'}) + \tau_2(-i\omega - E_n + E_{n'})} \times \tag{143}$$

$$e^{E_m\tau_{l_1} - E_{m'}\tau_{l_1+1} + E_n\tau_{l_2} - E_{n'}\tau_{l_2+1}} \times \tag{144}$$

$$\langle m_0| e^{-\int_0^{\tau_{l_1}} H(\tau)d\tau} |m\rangle A_{mm'} \langle m'| e^{-\int_{\tau_{l_1+1}}^{\tau_{l_2}} H(\tau)d\tau} |n\rangle B_{nn'} \langle n'| e^{-\int_{\tau_{l_2}}^{\beta} H(\tau)d\tau} |m_0\rangle \tag{145}$$

and after integration

$$\chi(i\omega) = \frac{1}{Z\beta}\sum_{l_1,l_2}\sum_{\{m\}} \frac{e^{i\omega\tau_{l_1+1} - E_m\Delta\tau_{l_1}} - e^{i\omega\tau_{l_1} - E_{m'}\Delta\tau_{l_1}}}{i\omega - E_m + E_{m'}} \tag{146}$$

$$\frac{e^{-i\omega\tau_{l_2+1} - E_n\Delta\tau_{l_2}} - e^{-i\omega\tau_{l_2} - E_{n'}\Delta\tau_{l_2}}}{-i\omega - E_n + E_{n'}} \times \tag{147}$$

$$\langle m_0| e^{-\int_0^{\tau_{l_1}} H(\tau)d\tau} |m\rangle A_{mm'} \langle m'| e^{-\int_{\tau_{l_1+1}}^{\tau_{l_2}} H(\tau)d\tau} |n\rangle B_{nn'} \langle n'| e^{-\int_{\tau_{l_2}}^{\beta} H(\tau)d\tau} |m_0\rangle$$

where $\Delta\tau_l = \tau_{l+1} - \tau_l$. This sum can be computed for each accepted diagram and sampled over Monte Carlo generated Markov chain. Since one needs to compute double sum over all kinks (l_1, l_2) 5-times sum over all atomic state $(\{m\})$ for each Matsubara frequency $i\omega$ it becomes increasingly expensive for the complicated atoms or low temperature with large number of kinks.

The response function dramatically simplifies in the case matrix elements $A_{mm'} = \delta_{mm'} A_m$ and $B_{nn'} = \delta_{nn'} B_n$ are diagonal and constant for each superstate (For example spin or charge susceptibiity). In this case, we can write

$$\chi(i\omega) = \frac{1}{Z\beta} \sum_{l_1,l_2} \sum_{\{m\}} \frac{e^{i\omega\tau_{l_1+1}} - e^{i\omega\tau_{l_1}}}{i\omega} \frac{e^{-i\omega\tau_{l_2+1}} - e^{-i\omega\tau_{l_2}}}{-i\omega} A_m B_n \times$$

$$\langle m_0|e^{-\int_0^{\tau_{l_1}} H(\tau)d\tau}|m\rangle e^{-E_m \Delta\tau_{l_1}} \langle m|e^{-\int_{\tau_{l_1+1}}^{\tau_{l_2}} H(\tau)d\tau}|n\rangle \times \quad (148)$$

$$e^{-E_n \Delta\tau_{l_2}} \langle n|e^{-\int_{\tau_{l_2}}^{\beta} H(\tau)d\tau}|m_0\rangle$$

The last part is just the probability for the atomic state $|m_0\rangle$, i.e., P_{m_0} therefore

$$\chi(i\omega) = \frac{1}{\beta} \sum_{m_0} P_{m_0} \sum_{l_1} \frac{e^{i\omega\tau_{l_1+1}} - e^{i\omega\tau_{l_1}}}{i\omega} A_{m_{l_1}} \sum_{l_2} \frac{e^{-i\omega\tau_{l_2+1}} - e^{-i\omega\tau_{l_2}}}{-i\omega} B_{n_{l_2}}$$

Here A_{m_l} is the value of the operator A in the interval $[\tau_l, \tau_{l+1}]$ provided the time evolution is started by state m_0.

If A and B are the equal operators, we finally have

$$\chi(i\omega) = \frac{1}{\beta}\sum_{m_0} P_{m_0}\left|\sum_{l_1}\frac{e^{i\omega\tau_{l_1+1}} - e^{i\omega\tau_{l_1}}}{i\omega}A_{m_{l_1}}\right|^2$$

and zero frequency is

$$\chi(0) = \beta\sum_{m_0} P_{m_0}\left|\sum_{l_1}\frac{\Delta\tau_l}{\beta}A_{m_{l_1}}\right|^2$$

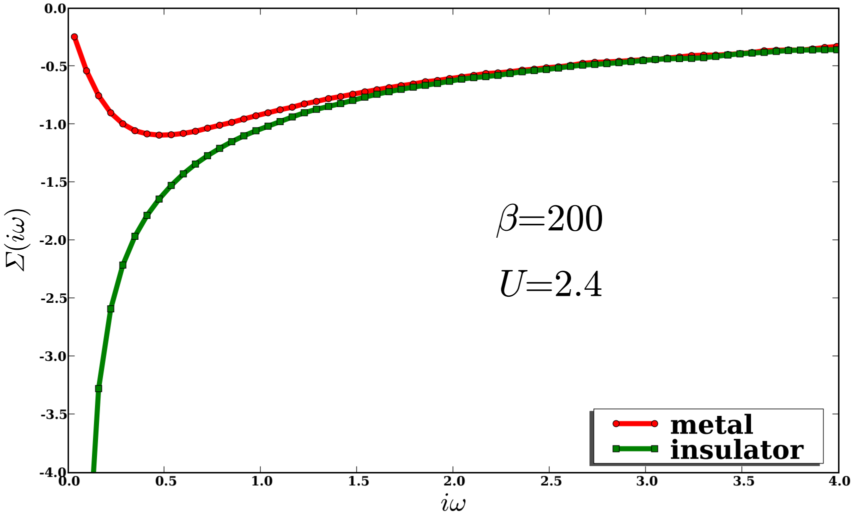
To compute this time evolution, we start with state m_0 where A takes value A_{m_0} and after first kink, the state becomes m_1 and A takes value A_{m_1},...

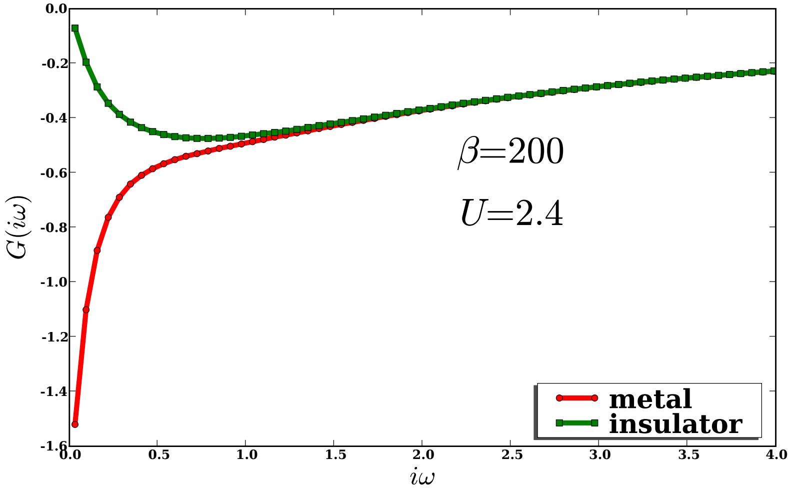
For zero frequency response functions, the term in the brackets is the average value of A in the imaginary time interval $[0,\beta]$ therefore one could symbolically write the above equation

$$\chi(0) \sim \langle|\frac{1}{\beta}\int_0^\beta A(\tau)d\tau|^2\rangle \tag{149}$$

which could be guessed from fluctuation-dissipation theorem.

Examples of the CTQMC run (Coexistance of metallic and insulating solution in the half-filled Hubbard model in $D \to \infty$:





Homework:

- Download the code from

  `http://www.physics.rutgers.edu/~haule/681/src_CTQMC/`

- Compile the source code in subdirectory `src`
- Check the script `iterate.py` in the `run` subdirectory. The script properly runs the code creating necessary input files.
- Change parameters U and mu to sketch the phase diagram of the one band Hubbard model in $D \to \infty$ limit. Since we are interested in the phase diagram of the half-filled model, mu should be always equal to $U/2$.
  - Verify that for $U = 2.4$ and $mu = 1.2$ and $beta = 100$ two different solutions are possible (coexistance of solutions)
  - Slowly increase U and check when the metallic solution disappears.
  - Slowly decrease U and check when insulating solution dissapears.
  - Sketch the region where the two solutions coexist.
  - Increase temperature and find the coexistence region for few temperatures.

Some explanation of the input to CTQMC and its use

The script `iterate.py` generates three input files for the impurity solver:

- `PARAMS`,
- *bath hybridization function* $\Delta(i\omega)$,
- *atomic energies* E_m *and matrix elements of* $\langle m|\psi^\dagger|n\rangle$.

A minimal `PARAMS` file can contain the following parameters (default values available)

- `Delta`– Filename of the input bath function hybridization function
- `cix` – Filename of the input file with atomic energies and matrix elements $\langle m|\psi^\dagger|n\rangle$.
- `U` – Coulomb repulsion (Slatter F0). Important remark: In cluster calculation, this parameter has to be set to zero because the onsite Coulomb repulsion needs to be entered through the eigenvalues of the atomic states.
- `mu` – The chemical potential
- `beta` – The inverse temperature
- `M` – The total number of qmc steps (every trial step counts!)
- `nom` – The number of imaginary frequency points treated by simulation
- `CleanUpdate`– Number of qmc steps between clean updates
- `aom` – To augment the high frequency tail of the self-energy with the low frequency part, one needs to compute the value of $\Sigma(i\omega_{max})$. If the statistics is very good, one can just take the last matsubara point. If not, aom counts the number of frequency points (to average over) to compute $\Sigma(i\omega_{max})$

More options include:

- Gf – ("Gf.out") Filename of the output green's function $G(i\omega)$
- Sig – ("Sig.out") Filename of the output self-energy $\Sigma(i\omega)$
- Ntau – (500) Number of imaginary time τ points to spline input hybridization function $\Delta(\tau)$.
- Nmax – (1024) Maximum number of kinks created during simulation (check the output "Histogram.dat" if this number is sufficiently large)
- PChangeOrder– (0.9) Probability to change the order (as compared to move a kink): the ratio between the step number $1\&2$ compared to step number 3.
- tsample – (50) How many qmc steps between two measurements.
- warmup – (500000) How many qmc steps before we start taking measurements
- CleanUpdate – (100000) How many steps before the clean update needs to be performed
- minM – (1e-10) Trace shuld always be larger than this minimal value when accepting the step (To avoid singular contributions)
- minD – (1e-10) Determinant of hybridization should always be larger than this number when accepting the step
- Ncout – (1000000) How often to output on the screen basic information about current state of the

simuation

- Naver – (500000000) How often to output much more information about simulation (for debugging purposes)
- TwoKinks – (0) Probability to try two kink step
- GlobalFlip – (-1) How often to try the global flip
- treshold – (1e-10) When computing the optimized atomic base, how small needs to be probability to neglect contributions of an atomic state
- SampleGtau – (-1) How often to update $G(\tau)$ by binning (not necessary since we have $G(i\omega)$ - used only for testing)
- sderiv – (0.1) Highest discrepancy in derivative allowed when matching the low frequency (simulated) and high frequency (moment expansion) part of self-energy.
- minDeltat – (1e-7) $\Delta(\tau)$ is sometimes not causal due to numerical error. In this case we set it to small value minDeltat.
- SampleSusc – (false) Spin and charge dynamic susceptibility are sampled in simulation
- nomb – (50) Number of bosonic frequencies when dynamic susceptibilities are computed
- som – (3) Similar to aom above but for suscpeptibility

The so called *cix* file (the precise name is given in the PARAMS input file) contains the information about the atomic states and how the atomic states are connected with conduction baths.

The baths and atomic states are an abstract concept and qmc code does not know to which particular quantum number a bath or an atomic states corresponds to. It is entirely responsibility of the user to properly input the information needed and to be consistent with the input. The QMC code does not know if the input is right and has no means to check that. The input file should be internaly consistent. We will explain below what each line of the input file means.

The input file starts with (lines with # are comments):

```
# Cix file for cluster DMFT with CTQMC
# cluster_size, number of states, number of baths, biggest ma
1 4 2 1
# baths, dimension, symmetry
0       1 0 0
1       1 0 0
```

The numbers in the first line are

- cluster size: For single impurity problem, cluster size is 1.
- number of all atomic states considered is here 4, namely $|0\rangle$, $|\downarrow\rangle$, $|\uparrow\rangle$, $|2\rangle$.
- number of bath components (here 2 because of spin $\uparrow$ and spin $\downarrow$). (Be careful: A multidimensional bath counts as one bath).
- The largest dimension of the matrix $\langle m|\psi^\dagger|n\rangle$.

In the next few lines, one needs to specify the dimension and the symmetry of each bath.

The four numbers in each line are:

- successive number of the bath (for example $\downarrow$ has index 0 and $\uparrow$ has index 1).
- The dimension of the bath
- The symmetry - if two baths are degenerate, they should have the same number. This index tells the code how to construct a matrix of $\Delta_{\alpha\beta}$ from the column in the input file `Delta.inp`. In this case, we have only one column (in addition to the column for the frequency) in the input. This collumn is coppied twice on diagonal when constructing a matrix Δ.
- The symmetry of the bath as given to the global flip subroutine. The global flip

subroutine will try to flip only baths for which this number is the same.

For more advanced example, we give 2×2 cluster of impurities in superconducting state:

```
# Cix file for cluster DMFT with CTQMC
# cluster_size, number of states, number of baths, maximum_ma
4 84 6 12
# baths, dimension, symmetry, global flip
  0    1       0                      0
  1    1     -0*                      1
  2    2       1     3    3   -1*     2
  3    2       1    -3   -3   -1*     3
  4    1       2                      4
  5    1     -2*                      5
```

The first line contains

- Now the cluster size is 4
- It turns out there are 84 superstates for 2×2 cluster.
- There are six baths in this case. How we choose them is explained below.

- The largest matrix $\langle m|\psi^\dagger|n\rangle$ has dimension 12×6 so we give 12.

All the local quantities (Green's function or hybridization or self-energy) are block diagonal (in properly choosen basis) and there are 6 independent blocks in Nambu formalism.The six lines above explain how are the three different input hybridization functions used to construct an 8×8 matrix of hybridizations.

- the integer number gives the column number in the input file of hybridization,
- $-$ means that the column has to be multiplied by -1 and
- $*$ means that the column needs to be complex conjugated.

The input file ”Delta.inp” should actually contain 9 columns, the first is frequency, than we have real and imaginary part of function named Δ_0, than real and imaginary part of function Δ_1,....

The hybridization matrix is constructed and, according to the input file, takes the following form

$$G = \left(\begin{array}{cc|cc|cc|cc}
\Delta_0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -\Delta_0^* & 0 & 0 & 0 & 0 & 0 & 0 \\
\hline
0 & 0 & \Delta_1 & \Delta_3 & 0 & 0 & 0 & 0 \\
0 & 0 & \Delta_3 & -\Delta_1^* & 0 & 0 & 0 & 0 \\
\hline
0 & 0 & 0 & 0 & \Delta_1 & -\Delta_3 & 0 & 0 \\
0 & 0 & 0 & 0 & -\Delta_3 & -\Delta_1^* & 0 & 0 \\
\hline
0 & 0 & 0 & 0 & 0 & 0 & \Delta_2 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -\Delta_2^*
\end{array}\right) \tag{150}$$

At the end of the simulation, the Green's function and self-energy take the exact same block diagonal form and the output Green's function contains again 9 coulmns.

In terms of quantum numbers for this particular case (cluster momentum $\mathbf{K}$ and spin σ), we can write the block structure of the green's function in the following way

$$G = \left(\begin{array}{cc|cc|cc|cc}
G_{(0,0)\uparrow}(i\omega) & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -G_{(0,0)\downarrow}(-i\omega) & 0 & 0 & 0 & 0 & 0 & 0 \\
\hline
0 & 0 & G_{(\pi,0)\uparrow}(i\omega) & G_{anomal}(i\omega) & 0 & 0 & 0 & 0 \\
0 & 0 & G^\dagger_{anomal}(i\omega) & -G_{(\pi,0)\downarrow}(-i\omega) & 0 & 0 & 0 & 0 \\
\hline
0 & 0 & 0 & 0 & \cdots & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & \cdots & \cdots & 0 & 0 \\
\hline
0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots
\end{array}\right) \tag{151}$$

The six baths, or equivalently, the six lines in the cix file, correspond to

- cluster momentum $\mathbf{K} = (0,0)$, spin ↑.
- cluster momentum $\mathbf{K} = (0,0)$, spin ↓.
- cluster momentum $\mathbf{K} = (\pi,0)$ forms a two dimensional block shown above
- cluster momentum $\mathbf{K} = (0,\pi)$ shows similar two dimensional block
- cluster momentum $\mathbf{K} = (\pi,\pi)$, spin ↑.
- cluster momentum $\mathbf{K} = (\pi,\pi)$, spin ↓.

From the above consideration, it is clear why the two central blocks have dimension 2 and the rest dimension 1.

Next line of the input file for the one-band model is

```
# cluster energies for unique baths, eps[k]
0
```

and for the cluster of 2×2 impurities is

```
# cluster energies for non-equivalent baths, eps[k]
-2 0 2  0
```

For the one-band model, we specified above that the two baths are equivalent (no breaking of spin symmetry). As a result only one impurity energy is needed. This impurity energy can be absorbed into the chemical potential and therefore is not needed in the input file.

In the case of the cluster, we have Δ_0, Δ_1, Δ_2 and Δ_3. The real part of these functions in infinity is in general not zero but approaches a constant. The input file, however, needs only the dynamic part of Δ while the static part needs to be entered here. This static part (or $\Delta(\omega = \infty)$) is the matrix of impurity levels.

Next few lines list all atomic eigenstates, there occupancy N, there momentum K, spin S_z, and dimension. For the one band model, we have

```
#   N   K    Sz  size
1   0   0     0    1      2  3      0   0
2   1   0  -0.5    1      0  4      0   0.5
3   1   0   0.5    1      4  0      0   0.5
4   2   0     0    1      0  0      0   0
```

The sixth line ocntains the index array $F^{0\dagger}(i)$, i.e., the operator of the bath 0 acts on the atomic state and leads to another atomic state. For example, $F^{0\dagger}(0) = 2$. The next, seventh column, contains similar index array, but for the bath number 1, i.e., $F^{1\dagger}(0) = 3$. It is important to understand how are these baths connected with the baths we specified above.

We explained above how the matrix of hybridizations is constructed from the input file. Each column (or row) in the hybridization matrix correspond to one bath (even if baths are equivalent). If the matrix of hybridizations takes the form

$$\left(\begin{array}{cc} \Delta_\uparrow & 0 \\ 0 & \Delta_\downarrow \end{array}\right) \tag{152}$$

the operator with index 0 is $\psi^\dagger_\uparrow$ and operator with index 1 is $\psi^\dagger_\downarrow$.

Finally, the last two columns stand for the energy of the atomic state and the total spin of the atomic state.

When writing the energies, we can excluded two trivial contributions to the atomic energies (but than we need to enter them through the PARAMS file):

- the shift in energies due to the chemical potential is trivially $-\mu N$ and is added later by the qmc code
- the splitting due to $F_0 \equiv U$ type of Coulomb repulsion adds to each atomic state an energy contribution $UN(N-1)/2$ and is also added by the qmc code. U is taken from the PARAMS file.

Much less trivial example of 2×2 cluster is given below:

```
 #   N   K   Sz size
 1   0   0    0   1      3   0   5   0   7   0   9   0                  0                  0
 2   1   0 -0.5   1     10   1  12   0  15   0  18   0                 -2                0.5
 3   1   0  0.5   1      0   0  13   0  16   0  19   0                 -2                0.5
 4   1   1 -0.5   1     12   0  10   1  18   0  15   0                  0                0.5
 5   1   1  0.5   1     13   0   0   0  19   0  16   0                  0                0.5
 6   1   2 -0.5   1     15   0  18   0  10   1  12   0                  0                0.5
 7   1   2  0.5   1     16   0  19   0   0   0  13   0                  0                0.5
 8   1   3 -0.5   1     18   0  15   0  12   0  10   1                  2                0.5
 9   1   3  0.5   1     19   0  16   0  13   0   0   0                  2                0.5
10   2   0    0   4     22   3  26   5  30   7  34   9     -3.10559821486                  0      2.44076863934      12.6
11   2   1   -1   2     25   4  21   2  33   8  29   6                 -2                  2                  1
12   2   1    0   4     26   5  22   3  34   9  30   7                 -2    -0.324555320337                  2      12.3
13   2   1    1   2     27   0  23   0  35   0  31   0                 -2                  2                  1
14   2   2   -1   2     29   6  33   8  21   2  25   4                 -2                  2                  1
.....
```

The firs few states written in direct base are

$$|1\rangle = |0,0,0,0\rangle$$

$$|2\rangle = \frac{1}{2}(|\downarrow,0,0,0\rangle + |0,\downarrow,0,0\rangle + |0,0,\downarrow,0\rangle + |0,0,0,\downarrow\rangle)$$

$$|3\rangle = \frac{1}{2}(|\uparrow,0,0,0\rangle + |0,\uparrow,0,0\rangle + |0,0,\uparrow,0\rangle + |0,0,0,\uparrow\rangle)$$

$$|4\rangle = \frac{1}{2}(|\downarrow,0,0,0\rangle - |0,\downarrow,0,0\rangle + |0,0,\downarrow,0\rangle - |0,0,0,\downarrow\rangle)$$

$$|5\rangle = \frac{1}{2}(|\uparrow,0,0,0\rangle - |0,\uparrow,0,0\rangle + |0,0,\uparrow,0\rangle - |0,0,0,\uparrow\rangle)$$

.....

There are 84 superstates and can be generated by the exact diagonalization code. Each atomic state has the following good quantum numbers $|i\rangle = |N, K, S_z\rangle$ which are written in the columns 2,3, and 4. The fifth column specifies how many eigenstates are combined into this particular superstate $|N, K, S_z\rangle$.

The cluster hybridization matrix written above is an 8×8 matrix. The column i in the hybridization matrix defines an operator $\psi^{i\dagger}$. In this case, the eight operators are enumerated so as to match the structure of the hybridization matrix, namely

$$\psi_1^\dagger = \psi_{00\uparrow}^\dagger \tag{153}$$

$$\psi_2^\dagger = \psi_{00\downarrow} \tag{154}$$

$$\psi_3^\dagger = \psi_{\pi 0\uparrow}^\dagger \tag{155}$$

$$\psi_4^\dagger = \psi_{\pi 0\downarrow} \tag{156}$$

$$\psi_5^\dagger = \psi_{0\pi\uparrow}^\dagger \tag{157}$$

$$\psi_6^\dagger = \psi_{0\pi\downarrow} \tag{158}$$

$$\psi_7^\dagger = \psi_{\pi\pi\uparrow}^\dagger \tag{159}$$

$$\psi_8^\dagger = \psi_{\pi\pi\downarrow} \tag{160}$$

The action of the above eight operators on each atomic state is specified in the columns 6-13. You can verify, for example

$$\psi_3^\dagger|2\rangle \equiv \psi_{\pi 0\uparrow}^\dagger|N=1, K=0, Sz=-1/2\rangle = |N=2, K=(\pi,0), Sz=0\rangle \equiv |12\rangle$$

Following the colum 13, the energies of all atomic states of a superstate are specified first, and then the total spin S for the same state. For example, if the state is two dimensional, one needs to specify two energies in columns 14 and 15 as well as spin S in columns 16 and 17. A two dimensional state thus needs 17 columns of input while the three dimensional state needs 19 columns...

Following the enumeration of the atomic states, we specify the matrix elements of the $\psi_i^\dagger$ between the blocks of two superstates. These matrices $(F^{\alpha\dagger})_{m'm} \equiv \langle j|\langle m'|\psi_\alpha^\dagger|m\rangle|i\rangle$ are very small and their size is equal to *dimension of superstate* i × *dimension of superstate* j. For the one band model, one has:

```
# matrix elements
1  2  1  1    1
1  3  1  1    1
2  0  0  0
2  4  1  1   -1
3  4  1  1    1
3  0  0  0
4  0  0  0
4  0  0  0
```

Each row of the above input specifies a matrix between the superstates. For each superstate and each bath, we use one row.

The first column specifies the starting atomic superstate $|i\rangle$, the second line the resulting atomic superstate $|j\rangle$ after appling operator $\psi_\alpha^\dagger$. The third and fourth column specify the dimension of both states. This is also the size of the matrix which follows. The full matrix between the superstates i and j follows and is input row by row in a single line, i.e.,

$$\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \rightarrow (1, 2, 3, 4) \tag{161}$$

Let us explain the first two rows. The first row says: applying operator $\psi^{1\dagger}$ to state $|1\rangle$ (first column), we get state $|2\rangle$ (second column). The dimension of the first atomic state is 1 (third column) and dimension of the second is 1 (fourth column) therefore the matrix is 1×1. The single matrix elements is specified in column five. The second line stands for bath number 2 and therefore explains: applying operator $\psi^{2\dagger}$ to state $|1\rangle$ (first column), we get state $|3\rangle$ (second column). The dimension of the first atomic state is 1 (third column) and dimension of the second is 1 (fourth column) therefore the matrix is 1×1. The single matrix elements is again specified in column five.

The less trivial example of the 2×2 cluster is written below

```
# matrix elements
 1   3    1   1                     1
 1   0    0   0
 1   5    1   1                     1
 1   0    0   0
 1   7    1   1                     1
 1   0    0   0
 1   9    1   1                     1
 1   0    0   0
 2  10    1   4        0.919901280179                    0       0.176804923698      -0.350030932463
 2   1    1   1                     1
 2  12    1   4        0.707106781187       0.573634850322                    0       0.413452607315
 2   0    0   0
 2  15    1   4        0.707106781187       0.573634850322                    0       0.413452607315
 2   0    0   0
 2  18    1   4                  -0.5                 -0.5                 -0.5                 -0.5
 2   0    0   0
 3   0    0   0
 3   0    0   0
 3  13    1   2                     1                    0
 3   0    0   0
 3  16    1   2                     1                    0
 3   0    0   0
 3  19    1   2       -0.707106781187      -0.707106781187
 3   0    0   0
 4  12    1   4       -0.707106781187       0.573634850322                    0       0.413452607315
......
```

The rest of the input file specifies how is the high-frequency tail of the self-energy treated. If $HB1$ word immediate follows matrix elements, we take Hubbard I-like scheme for the high-frequency expansion (using qmc probabilities in the expansion) which is exact to the second moment of the self-energy.

The states and matrix elements need to be repeated ones more. The reason is that one could project out some states in the original input when the low-frequency data is sampled. For high-frequency all states contribute and one needs to input here *more* atomic states in case truncation of atomic states was performed above.

There is an alternative scheme available, namely entering the high-frequency expansion of the self-energy by an arbitrary perl expression which can contain arbitrary complicated operators as long as the matrix elements of the operator are specified in the cix file. This scheme is seldom needed.