Continuous Time Quantum Monte Carlo (CTQMC)

Corey Melnick (cmelnick@bnl.gov)

Brookhaven National Laboratory

comscope
from codes to spectroscopies

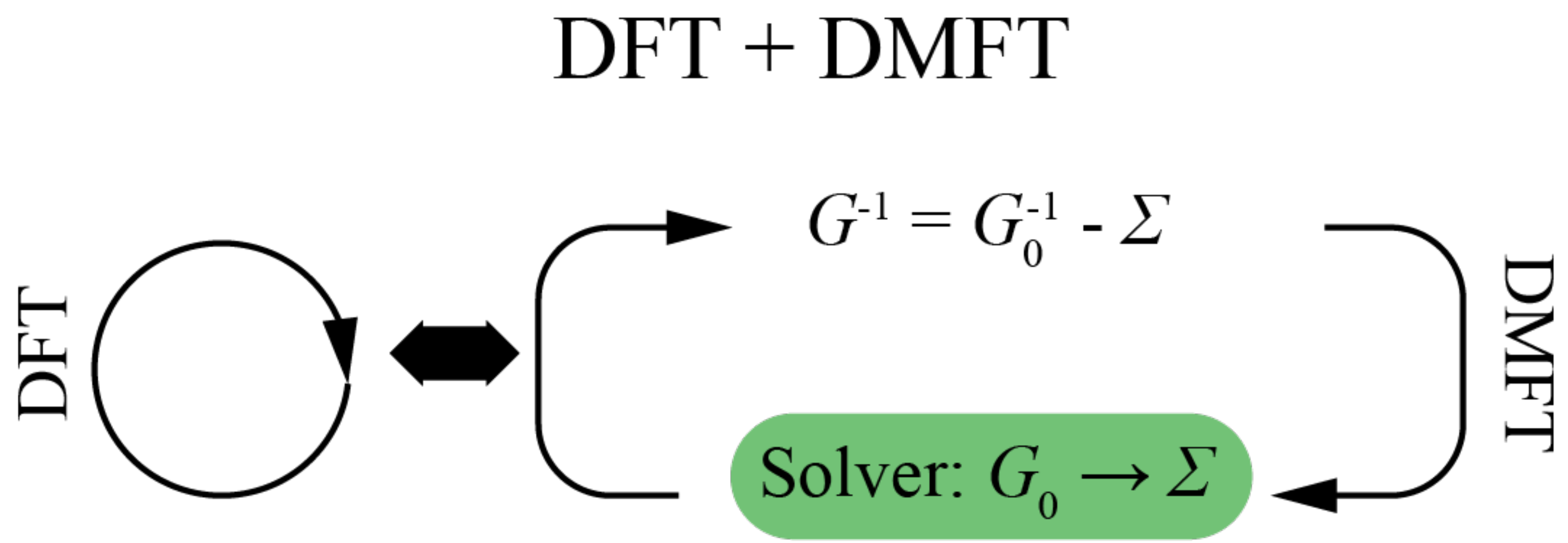
Outline

1. Context: Why are we talking about CTQMC?
2. Background: What is the quantum impurity problem?
3. CTQMC: How do we solve the quantum impurity problem with CTQMC?
4. Observables: How do we compute the self-energy needed by DMFT?
5. Limitations: What can't we do with CTQMC?
6. Questions

1. Context

Continuous Time Quantum Monte Carlo (CTQMC)

- Solves quantum impurity problems, e.g., the heart of the DMFT equations



DMFT: A. Georges, G. Kotliar, W. Krauth, and M. J. Rozenberg, *Rev. Mod. Phys.*, vol. 68, p. 13, 1996.
DFT+DMFT: G. Kotliar *et al.*, *Rev. Mod. Phys.*, vol. 78, no. 3, pp. 865–951, 2006.
Solver (CTQMC): E. Gull *et al.*, *Rev. Mod. Phys.*, vol. 83, no. 2, pp. 349–404, 2011.

1. Context

Solvers

Exact	Approximate
Numerical renormalization group (NRG)	One-crossing approximation (OCA)
Exact Diagonalization (ED)	Hubbard-one
CTQMC	

Why CTQMC?

- Only exact quantum impurity solver which handles real materials
- Extremely parallelizable (near ideal scaling)
- GPU accelerated (up to 225x for f-shell problems)

2. The Quantum Impurity problem

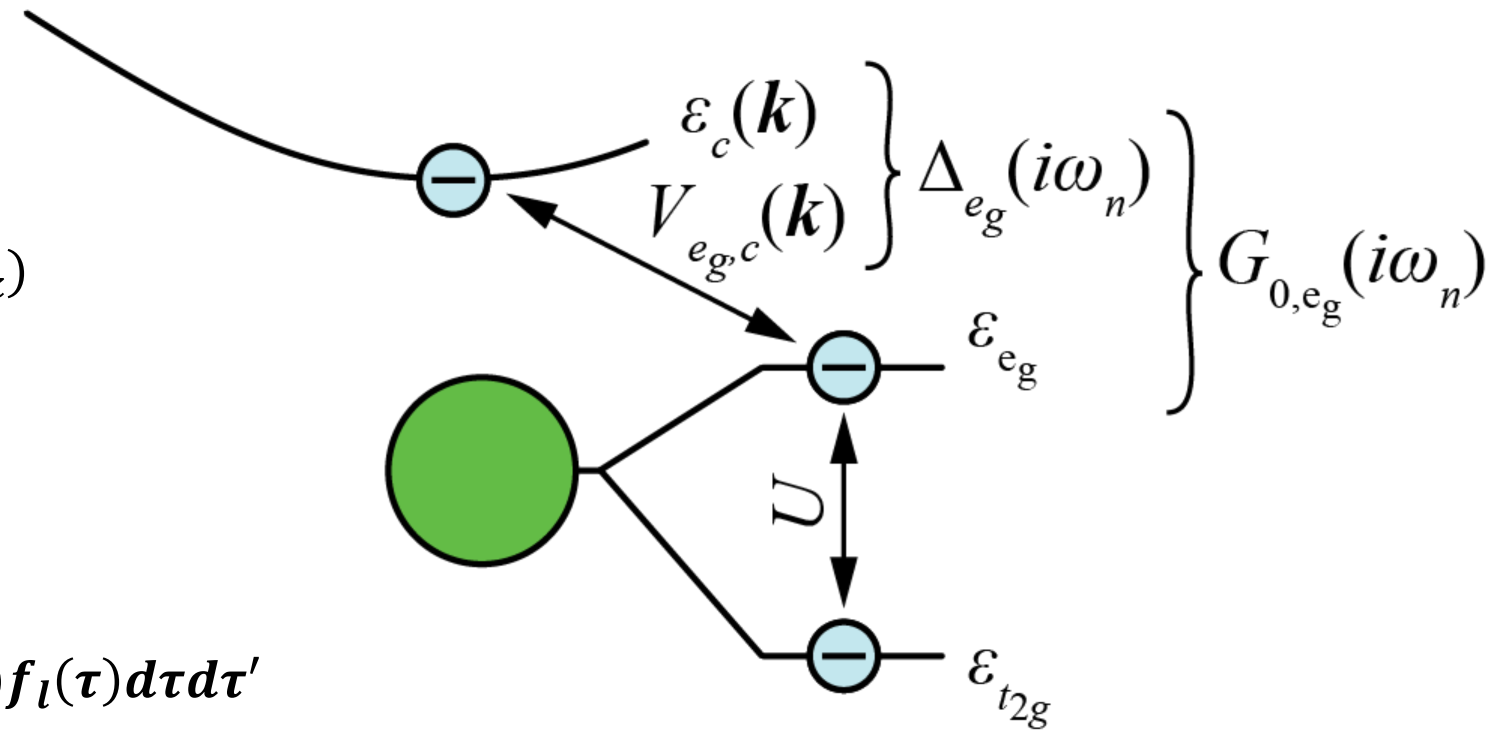
Anderson Impurity Model

- Hamiltonian:

$$\boldsymbol{H_{AIM}} = \sum_k \boldsymbol{\epsilon_k c_k^\dagger c_k} + \boldsymbol{\epsilon_f f^\dagger f} + \boldsymbol{U n_\uparrow n_\downarrow} + \sum_k (\boldsymbol{V_k c_k^\dagger f} + \boldsymbol{V_k^* f^\dagger c_k})$$

- Action:

$$\boldsymbol{S} = \sum_{ij} \iint \boldsymbol{f_i^\dagger(\tau) \mathcal{G}_{0,ij}^{-1}(\tau - \tau') f_j(\tau') d\tau d\tau'} + \sum_{ijkl} \iint \boldsymbol{f_i^\dagger(\tau) f_j^\dagger(\tau') \mathcal{U}_{ijkl}(\tau - \tau') f_k(\tau') f_l(\tau) d\tau d\tau'}$$

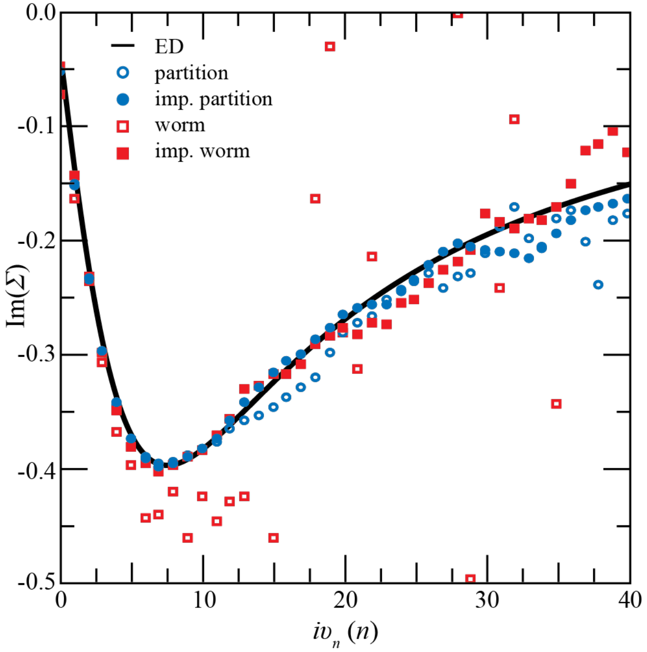


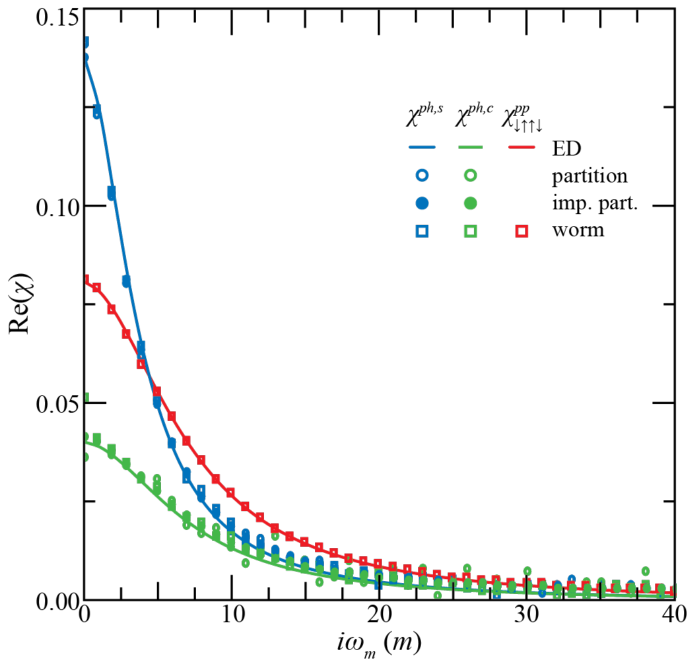
➢ A dynamical mean-field and a (dynamical) interaction

2. The Quantum Impurity problem: Solution

What does it mean to solve an impurity problem?

- Correlation and vertex functions
  - $\boldsymbol{G_{ij}(\tau-\tau') = -\langle T_\tau f_i(\tau) f_j^\dagger(\tau')\rangle}$
  - $\boldsymbol{\Sigma = \mathcal{G}_0^{-1} - G^{-1}}$
  - $\boldsymbol{G_{ijkl}(\tau-\tau') = -\left\langle T_\tau f_i(\tau) f_j^\dagger(\tau) f_k(\tau') f_l^\dagger(\tau')\right\rangle}$
  - $\boldsymbol{\Gamma = \chi_0^{-1} - \chi^{-1}}$





C. Melnick *et al.*, "Accelerated impurity solver for DMFT and its diagrammatic extension" *arxiv* 2010.08482 (2020)

3. CTQMC: The Idea

1. Split the Hamiltonian into two parts

$$H = H_a + H_b$$

2. Expand the partition function in orders of H_b

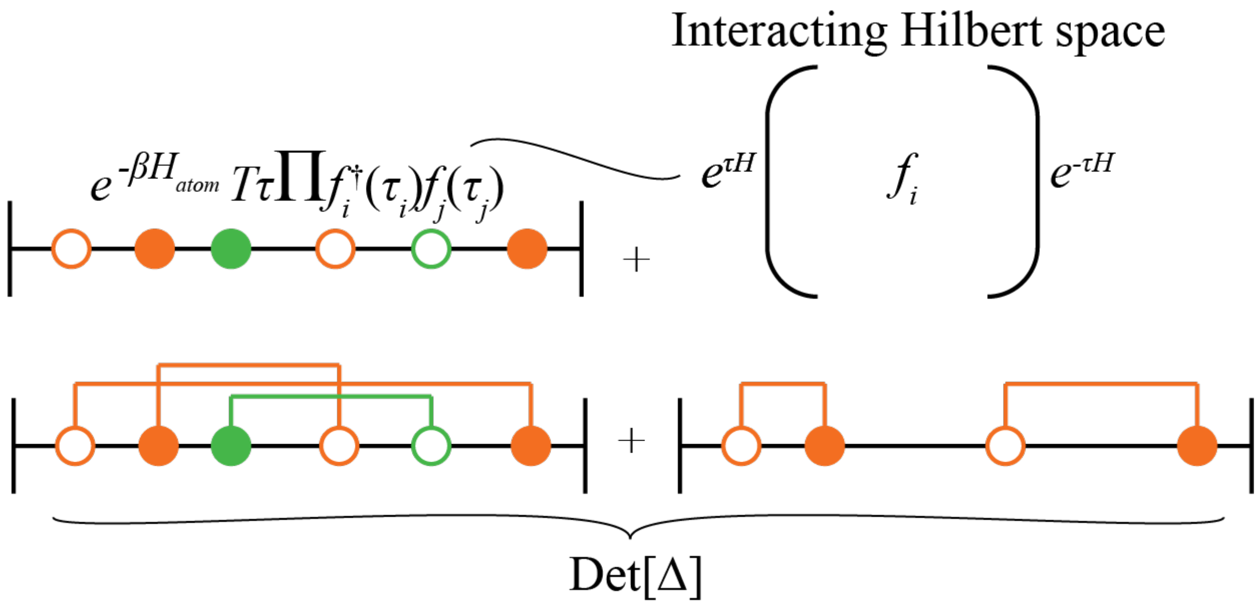
$$Z = \int \mathcal{D}[f, f^\dagger] e^{-S} = Tr\, T_\tau e^{-\beta H_a} exp\left[-\int d\tau H_b(\tau)\right]$$

$$= \sum_k (-1)^k \int_0^\beta d\tau_1 \dots \int_{\tau_{k-1}}^\beta d\tau_k\, Tr\left[e^{-\beta H_a} H_b(\tau_k) \dots H_b(\tau_1)\right]$$

$$= \sum_{k\gamma} \int_0^\beta d\tau_1 \dots \int_{\tau_{k-1}}^\beta d\tau_k w(k, \gamma, \tau_1, \dots., \tau_k)$$

- H_a is something "easy" to compute

- Example: (CT-HYB)
  - *a)* H_a : Local (atomic) Hamiltonian
  - *b)* H_b : Hybridization functions



3. CTQMC: How do we sample the expansion?

1. We have integral of weights for diagrams of expansion order k

$$Z = \sum_{k=0}^{\infty} \sum_{\gamma} \int_0^{\beta} d\tau_1 \ldots \int_{\tau_{k-1}}^{\beta} d\tau_k \, w(k, \gamma, \tau_1, \ldots, \tau_k)$$

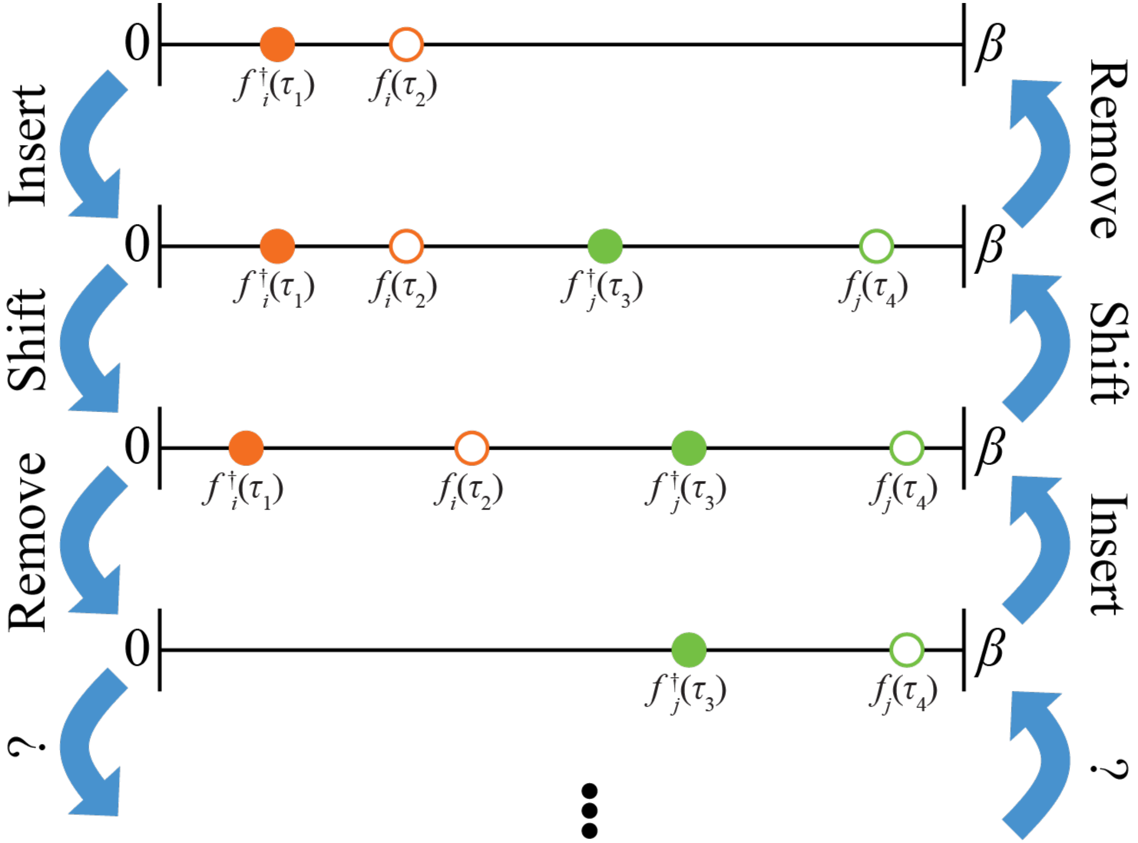
2. Create a Markov Chain of diagrams
   a. Propose updates to the current diagram
      - Insert or remove vertices

   b. Metropolis-Hastings

   $$R_{x \to y} = \frac{w(y) W_{yx}}{w(x) W_{xy}}$$

   c. accept or reject the move

   $$r < min(1, R_{x \to y})$$



3. CTQMC: CT-HYB

Within the hybridization expansion, we take the atomic Hamiltonian as H_a, and the hybridization as H_b

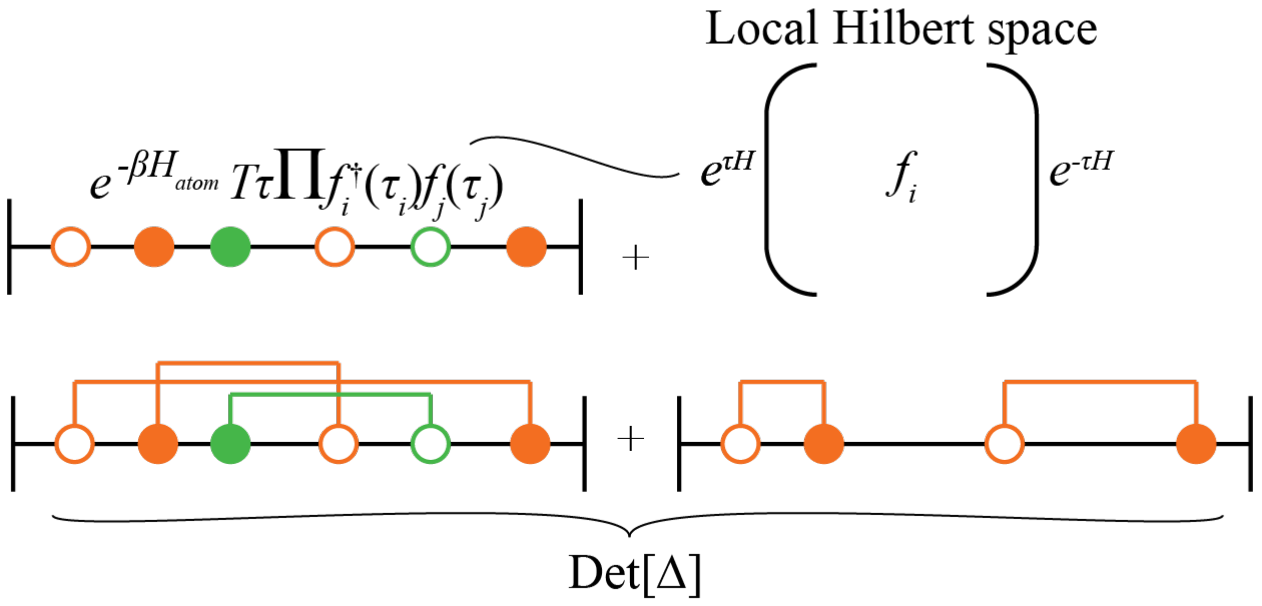
$$w(k,\gamma,\tau_1,\ldots,\tau_k) = w(x) = w_{loc}w_{hyb}$$

1. The local impurity trace

$$w_{loc}(x) = \mathrm{Tr}\, e^{-\beta H_{loc}} T_\tau \prod_r^k f_{i'_r}(\tau'_r) f^\dagger_{i_r}(\tau)$$

2. The hybridization matrix determinant

$$w_{hyb}(x) = \prod_r^k \Delta_{i_r i'_r}(\tau_r - \tau'_r) = \mathrm{Det}\Delta$$



4. Observables: The idea

- Now we need to figure out how to compute the desired observables

$$G_{ij}(\tau-\tau') = -\langle T_\tau f_i(\tau) f_j^\dagger(\tau')\rangle$$

$$G_{ijkl}(\tau-\tau') = -\langle T_\tau f_i(\tau) f_j^\dagger(\tau) f_k(\tau') f_l^\dagger(\tau')\rangle$$

- Just as we wrote an expansion for the partition function, we can write an expansion for the local observable

$$\langle O\rangle = Z^{-1}\int \mathcal{D}[f,f^\dagger] e^{-S}\, O$$

- We use this to accumulate an estimate of the observable as we sample Z

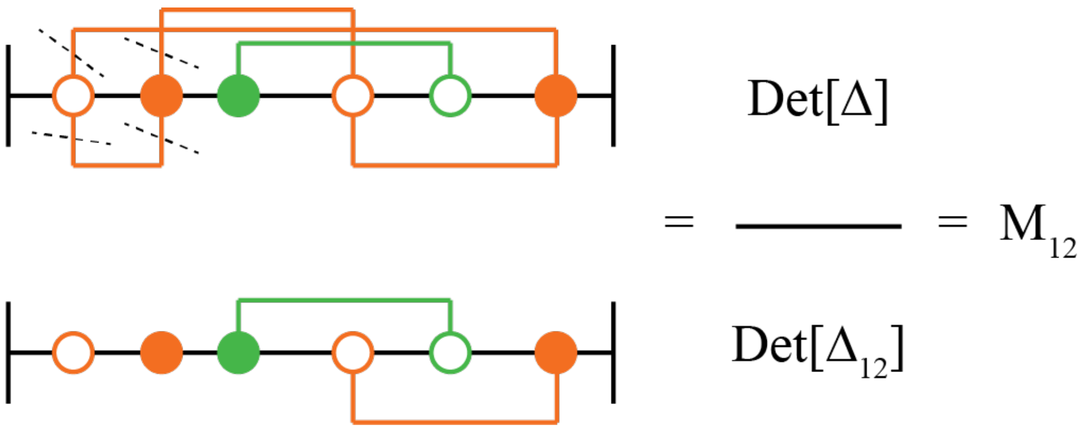
$$\langle O\rangle = Z^{-1}\sum_x w(x,O) = Z^{-1}\sum_x w(x)\frac{w(x,O)}{w(x)} = Z^{-1}\sum_x w(x)\, o(x,O)$$

4. Observables: Example

- Consider the one-particle green's function

$$\langle G_{ij}(\tau-\tau')\rangle = Z^{-1}\sum_x w(x)\frac{w(x,\tilde{f}_i(\tau)\tilde{f}_j^\dagger(\tau'))}{w(x)}$$

  - It is easier to remove hybridization lines than it is to insert new operators
    - k-2 measurements
    - Precomputed weights



- $\langle G_{ij}(\tau-\tau')\rangle = Z^{-1}\sum_x w(x)\frac{w(x,\tilde{f}_i(\tau)\tilde{f}_j^\dagger(\tau'))}{w(x)} = \frac{1}{N}\sum_n^N[\sum_{rs}\delta_{\tau-\tau,\tau_r-\tau'_s}\delta_{ii'_s}\delta_{ji_r}\mathbf{M}_{\mathbf{sr}}^{-1}]_n$

4. Observables: Practicalities and Limitations

- Better basis sets (Matsubara, Legendre or other, more advanced, ideas)

$$\langle G_{ij}(i\omega_n)\rangle = \left\langle \sum_{rs} e^{i\omega_n(\tau'_s-\tau_r)}\,\delta_{ii'_s}\delta_{ji_r}\Delta^{-1}_{\mathrm{sr}} \right\rangle$$

$$\langle G_{ij}(l)\rangle = \left\langle \sum_{rs} P_l(\tau'_s-\tau)\,\delta_{ii'_s}\delta_{ji_r}\Delta^{-1}_{\mathrm{sr}} \right\rangle$$

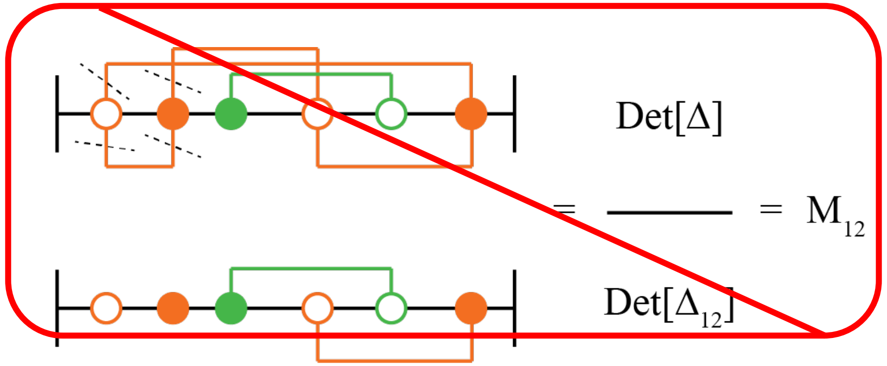
- We cannot measure when
  - $w(x) \to 0$ and $w(x, O) \not\to 0$
    - You can't measure G_{ij} unless Δ_{ij} is non-zero
    - No big deal in DFT+DMFT: $\Delta_{ij} = 0$ implies $G_{ij} = 0$
  - Big issue in GW+DMFT:
    - $G_{ijkl} \neq 0$ when $\Delta_{ij} = 0$ or $\Delta_{kl} = 0$
    - We can typically only measure G_{iijj} components in a similar manner to G_{ij}!

Brookhaven National Laboratory

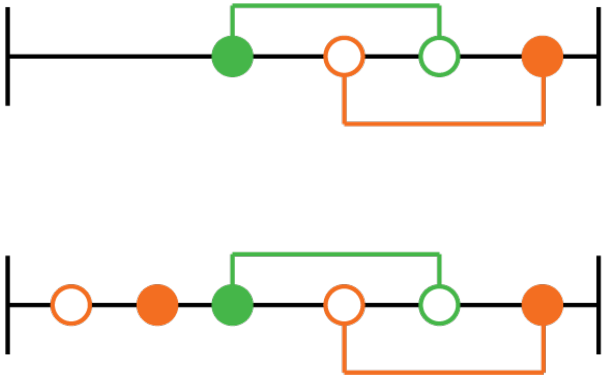
L. Boehnke, H. Hafermann, M. Ferrero, F. Lechermann, and O. Parcollet, "Orthogonal polynomial representation of imaginary-time Green's functions," Phys. Rev. B, vol. 84, no. 7, p. 75145, Aug. 2011

4. Observables: The Worm Algorithm

- Recall
  - *"It is easier to remove hybridization lines"*



- Add to local operators into configuration using Metropolis-Hastings



$$= \frac{\mathrm{Tr}[f_g^\dagger(\tau_1) f_o(\tau_2) f_g(\tau_3) f_o^\dagger(\tau_4)]}{\mathrm{Tr}[f_o(\tau') f_o^\dagger(\tau'') f_g^\dagger(\tau_1) f_o(\tau_2) f_g(\tau_3) f_o^\dagger(\tau_4)]} = \frac{w_{\mathrm{loc}}(x)}{w_{\mathrm{loc}}(x,O)}$$

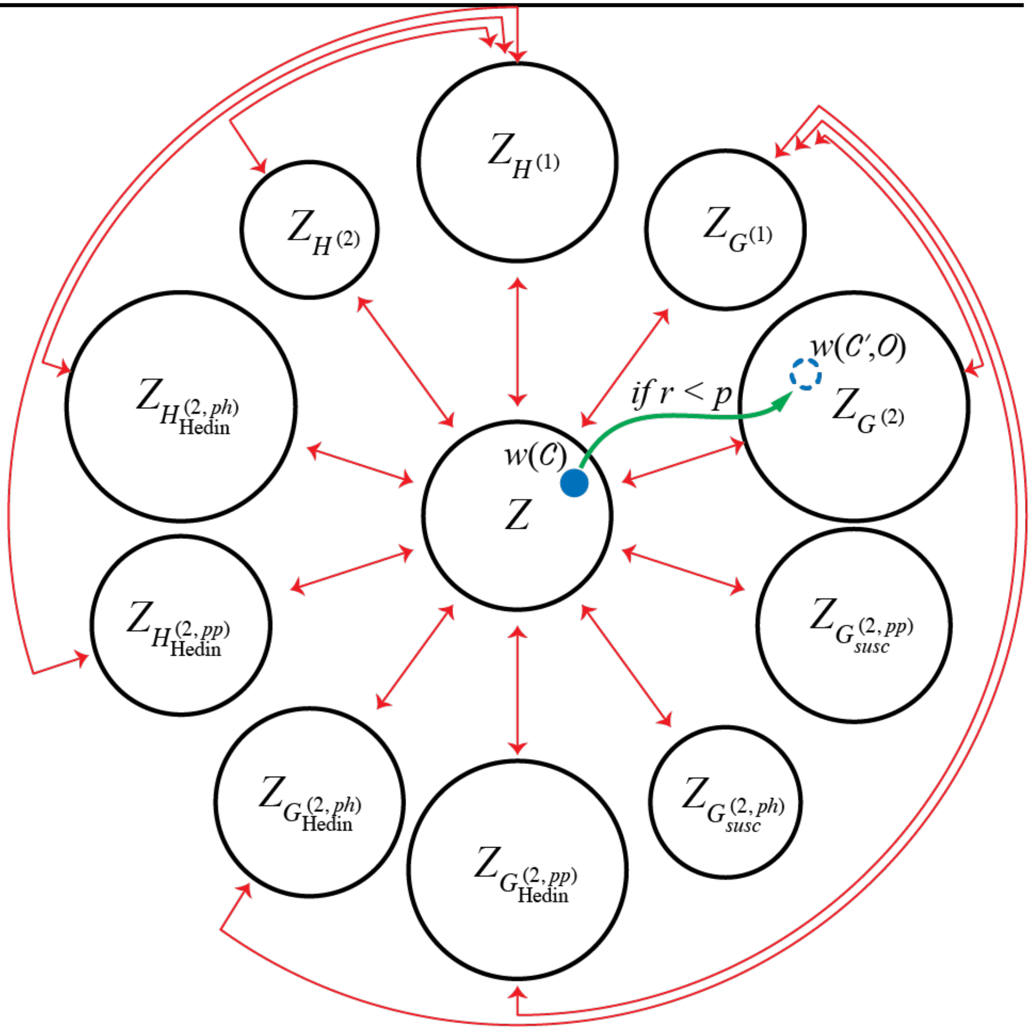
  - "Worm" from partition space into Green's function space (or any observable space!)

Brookhaven National Laboratory

P. Gunacker *et al*. "Continuous-time quantum Monte Carlo using worm sampling," *Phys. Rev. B*, vol. 92, no. 15, p. 155102, 2015

4. Observables: Worm Spaces

- With this idea, we can sample any correlator "easily"
  - The more operators and times, the more samples we require
  - Relative sizes and Wang-Landau



Brookhaven National Laboratory

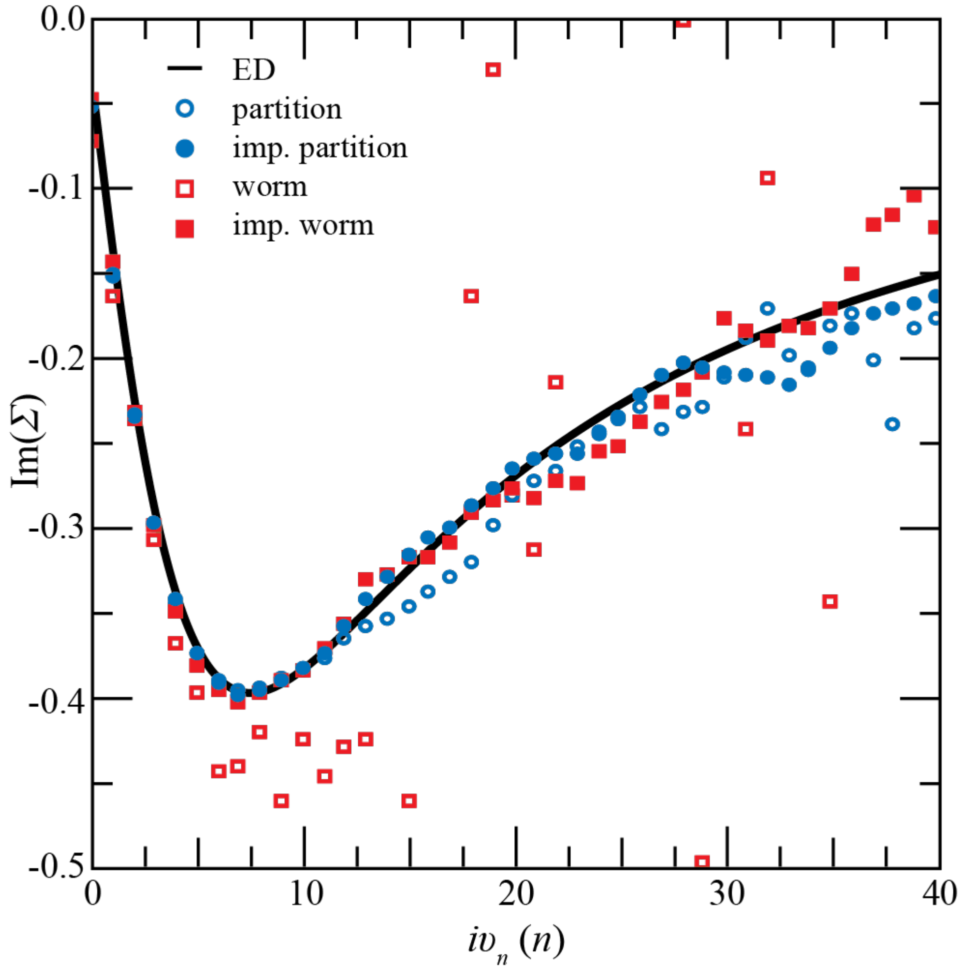
C. Melnick *et al.*, "Accelerated impurity solver for DMFT and its diagrammatic extension" *arxiv* 2010.08482 (2020)

4. Observables: Worm Measurements

- Measurement is trivial!
  - Just count how often we wind up in a particular Green's function space

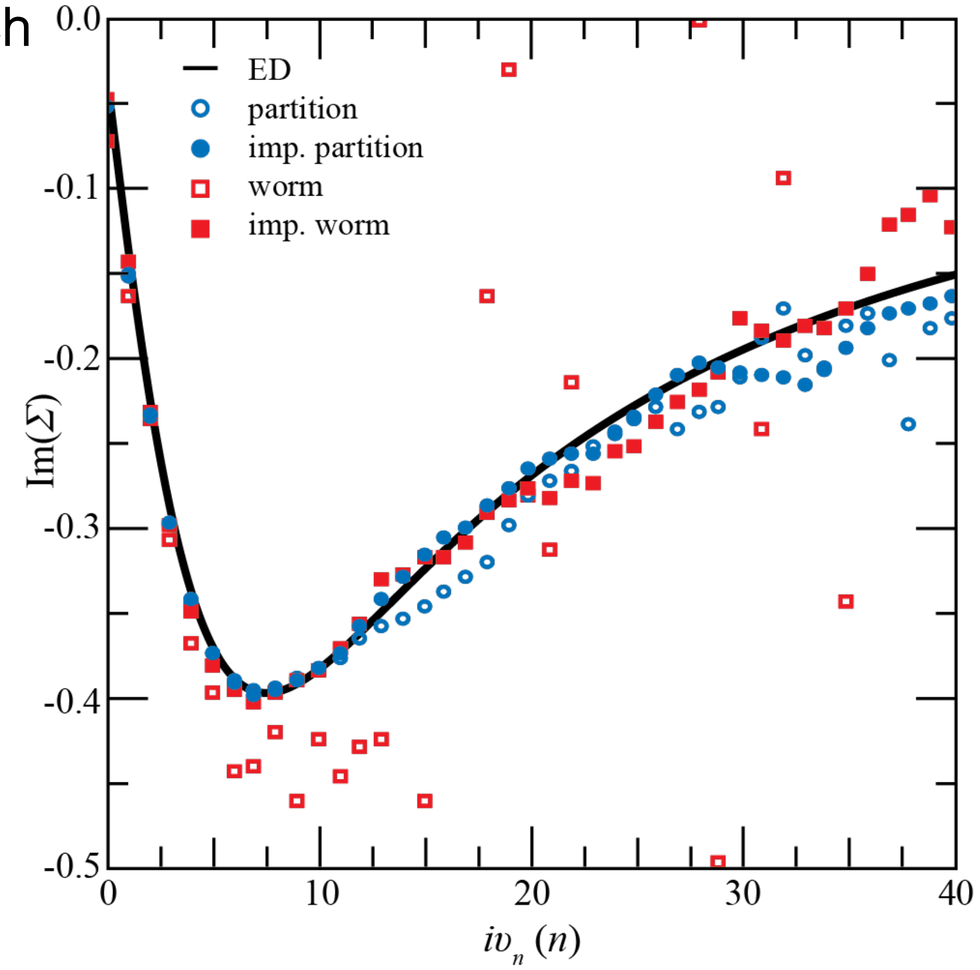
$$\langle G_{ij}(i\omega_n)\rangle = \left\langle e^{i\omega_n(\tau'-\tau)}\right\rangle_{G_{ij}}$$

$$\langle G_{ijkl}(i\omega_n)\rangle = \left\langle e^{i\omega_n(\tau'-\tau)}\right\rangle_{G_{ijkl}}$$



4. Observables: Improved Estimators

- Briefly mentioned that Vertex functions converge much slower than correlation functions
  - $G_{ij}(\tau - \tau') = -\langle T_\tau f_i(\tau) f_j^\dagger(\tau') \rangle$
  - $\Sigma = \mathcal{G}_0^{-1} - G^{-1}$
  - $\delta\Sigma = \mathrm{G}^{-2}\delta G \propto \omega_n^2 \delta G$
- It is much better to sample $\mathbf{G\Sigma}$
  - $\Sigma = G^{-1} G\Sigma$
  - $\delta\Sigma = \mathrm{G}^{-2} G\Sigma\delta G + G^{-1}\delta(G\Sigma) \propto \omega_n \delta G \delta(G\Sigma)$
- Improved estimators can be generated from the equations of motion
  - $\mathrm{G\Sigma} = -\langle T_\tau [f_i, U](\tau) f_j^\dagger \rangle$
  - $\mathrm{H} = -\langle T_\tau [f_i, U](\tau) f_j^\dagger f_k f_l^\dagger \rangle$



H. Hafermann *et al.*, Phys. Rev. B., vol. 85, no. 20, pp. 1–14, 2012
P. Gunacker *et al.*, Phys. Rev. B, vol. 94, p. 125153, 2016.

5. Limitations: The sign problem

- We have ignored a big problem for CTQMC: the infamous sign problem!
- Consider the Metropolis-Hastings algorithm

$$R_{x\to y} = \frac{w(y)W_{yx}}{w(x)W_{xy}}$$

  - This only works if $w(x) \geq 0$
  - But we are working with Fermions so $w(x) \leq 0$ is quite possible
  - We make the following adjustment

$$R_{x\to y} = \frac{|w(y)|W_{yx}}{|w(x)|W_{xy}}$$

  - We are no longer sampling Z, we are sampling $|Z|$!
  - So, we must adjust our observable estimators

$$\langle O \rangle = \frac{|Z|}{Z} \langle \frac{w(x)}{|w(x)|} o(x, O) \rangle$$

5. Limitations: Computational Cost of the Sign Problem

- Examining our expression for the observable

$$\langle \boldsymbol{O} \rangle = \frac{|\boldsymbol{Z}|}{\boldsymbol{Z}} \langle \frac{\boldsymbol{w}(\boldsymbol{x})}{|\boldsymbol{w}(\boldsymbol{x})|} \boldsymbol{o}(\boldsymbol{x}, \boldsymbol{O}) \rangle$$

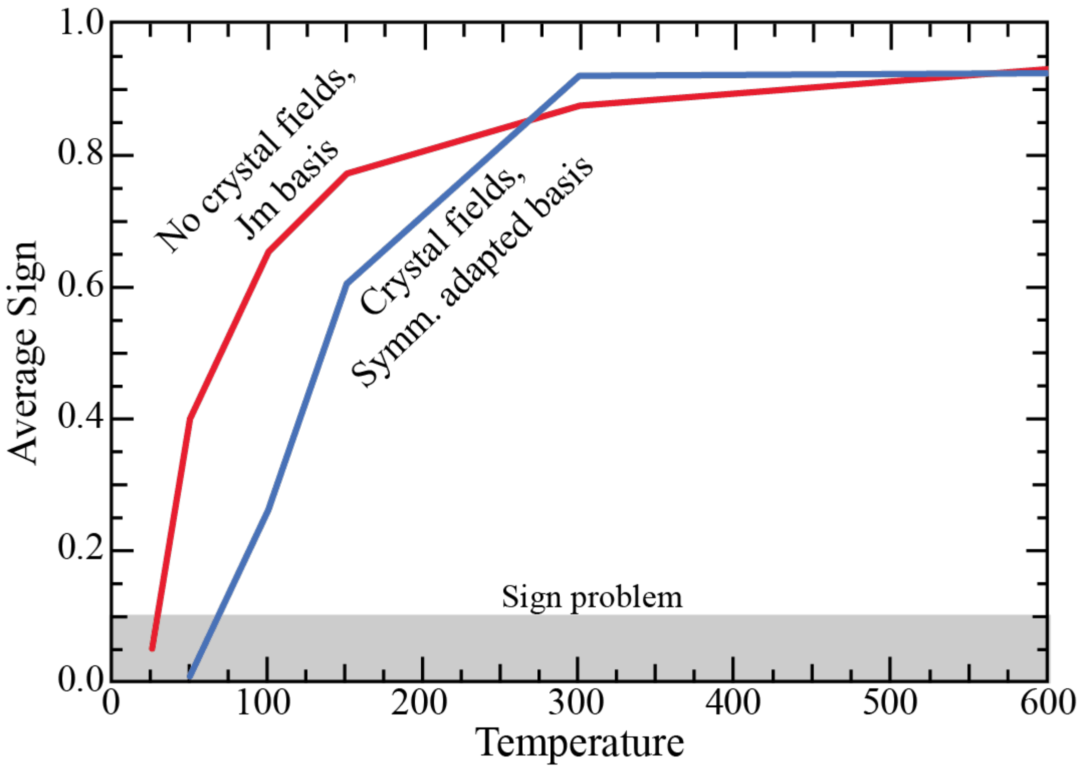
- Let us call $\frac{\boldsymbol{Z}}{|\boldsymbol{Z}|}$ the average sign, $\frac{\boldsymbol{w}(\boldsymbol{x})}{|\boldsymbol{w}(\boldsymbol{x})|}$, of the simulation

$$\frac{\boldsymbol{Z}}{|\boldsymbol{Z}|} = \langle \frac{\boldsymbol{w}(\boldsymbol{x})}{|\boldsymbol{w}(\boldsymbol{x})|} \rangle$$

- As the sign vanishes, $\frac{\boldsymbol{Z}}{|\boldsymbol{Z}|} \to \mathbf{0}$
  - We are collecting data which barely affects the estimator
  - More and more samples are required to converge the estimate
  - Computational requirements explode as 1/sign

5. Limitations: Behavior of the Sign Problem

- What exacerbates the sign problem?
  - Low temperature:
    - $\mathbf{T} \to \mathbf{0}$: sign $\to \mathbf{0}$
  - Off-diagonal elements in the hybridization matrix
    - $\boldsymbol{\Delta_{ij}} \neq \mathbf{0}$ for $\boldsymbol{i} \neq \boldsymbol{j}$
- What Helps?
  - Ising interactions only!
    - $\boldsymbol{U_{ijkl}} = \mathbf{0}$ for $\boldsymbol{i} \neq \boldsymbol{j}, \boldsymbol{k} \neq \boldsymbol{l}$ or $\boldsymbol{i} \neq \boldsymbol{l}, \boldsymbol{j} \neq \boldsymbol{k}$
  - Basis
    - Example: Fe at 600 K
      - Relativistic basis: sign < 0.1
      - Cubic harmonics: sign > 0.9
    - Example: delta-Pu (figure)
      - Symmetry adapted basis: *T* > 100 K
      - J-basis: *T* > 40 K



Brookhaven National Laboratory

comscope from codes to spectroscopies

5. Limitations: Computational Bottleneck

- Local impurity trace

$$w_{\text{imp}}(\mathrm{C}) = \mathrm{Tr}\, e^{-\beta H_{loc}} T_\tau \prod_r^k c_{i_r'}(\tau_r') c_{i_r}^\dagger(\tau_r) = \mathrm{Tr}\, P_{\beta-\tau_k} F_{i_k}^\dagger P_{\tau_k-\tau_k'} F_{i_k'} \dots F_{i_1} P_{\tau_1-\tau_1'} F_{i_1}^\dagger P_{\tau_1'}$$

where $(F_i)_{mn} = \langle m|c_i|n\rangle, P_\tau = e^{-\tau H_{loc}}$

- $\boldsymbol{F_i}$ is a matrix of rank 2^n
  - n is the number of orbitals (d: n =10 or f: n =14)
  - ➢ Computation is of order $O[k(2^n)^3]$
    - Even with GPUs, this is prohibitive
- Decompose Hilbert space $\mathscr{H}$ into sectors according to the Abelian symmetries, each of which has its own unique set of quantum numbers (N, S_z, etc.)

$$\mathscr{H} = \bigoplus_{\boldsymbol{q}=1}^{\boldsymbol{N}} \mathscr{H}(\mathbf{q})$$

$$[F_i(q_j)]_{mn} = \langle m(q_{j+1})|c_i|n(q_j)\rangle$$

  - These matrices are of much smaller rank!
- Store sub-products: O(k) → O(log k)

P. Sémon, C.-H. Yee, K. Haule, and A.-M. S. Tremblay, Phys. Rev. B, vol. 90, p. 75149, 2014.

5. Limitations: Overcoming Bottlenecks

- For smaller problems and at low temperature, computing the hybridization weights is the bottleneck, which requires taking the ratio of determinants: $O(k^2)$

- GPU's can handle the multiplication of many large matrices very well
  - We have achieved 15 - 225x acceleration of a Summit node for Plutonium problems (depending on the details)!

- CTQMC is massively parallelizable
  - Each Markov chain is entirely independent
  - Communication only at beginning and end
  - As long as measurement phase is long, scaling is ideal!
  - We've run ComCTQMC on 1000 nodes on Summit at 95% of the ideal.
    - Petascale!

6. Questions?