An Amelioration for the Sign Problem: Adiabatic Quantum Monte Carlo

Mohammad-Sadegh Vaezi,[1] Amir-Reza Negari,[2] Amin Moharramipour,[2] and Abolhassan Vaezi[2, *]

[1]*Pasargad Institute for Advanced Innovative Solutions (PIAIS) , Tehran 19916-33361, Iran*
[2]*Department of Physics, Sharif University of Technology, Tehran 14588-89694, Iran*

We introduce the adiabatic quantum Monte Carlo (AQMC) method, where we gradually crank up the interaction strength, as an amelioration of the sign problem. It is motivated by the adiabatic theorem and will approach the true ground-state if the evolution time is long enough. We demonstrate that the AQMC enhances the average sign exponentially such that low enough temperatures can be accessed and ground-state properties probed. It is a controlled approximation that satisfies the variational theorem and provides an upper bound for the ground-state energy. We first benchmark the AQMC vis-à-vis the undoped Hubbard model on the square lattice which is known to be sign-problem-free within the conventional quantum Monte Carlo formalism. Next, we test the AQMC against the density-matrix-renormalization-group approach for the doped four-leg ladder Hubbard model and demonstrate its remarkable accuracy. As a nontrivial example, we apply our method to the Hubbard model at $p = 1/8$ doping for a 16×8 system and discuss its ground-state properties. We finally utilize our method and demonstrate the emergence of $U(1)_2 \sim SU(2)_1$ topological order in a strongly correlated Chern insulator.

Introduction.— Quantum Monte Carlo (QMC) is one of the most powerful tools in computational physics [1–6]. It maps an interacting problem into an ensemble over infinitely many non-interacting problems through the Hubbard-Stratonovich (HS) transformations [7]. Every space-time realization of the HS fields defines a path integral which can be evaluated exactly. For fermionic models, it usually amounts to computing a determinant. The Metropolis-Hastings algorithm is then employed to sample the HS fields according to their weights, namely the corresponding path integrals [8, 9]. These weights are in general not positive definite and their signs can fluctuate strongly. It has been shown that for a system with volume V and at temperature T: $\langle \text{sign} \rangle \propto \exp(-fV/T)$, where f is a model-dependent constant with units of free energy density [10]. Furthermore, the number of samplings required to achieve a desired accuracy scales with the average sign (of the weights) as $1/\langle \text{sign} \rangle^2$. Therefore, except for a restricted class of models which are guaranteed to have positive weights [11–23], the sign problem limits the applicability of the QMC approach to high temperatures and small systems [10, 24].

Since the sign problem is NP-hard [10] and does not have a general solution, it can be at best alleviated [25–29]. In this letter, we introduce the adiabatic quantum Monte Carlo (AQMC) algorithm as a novel tool to mitigate the sign problem and boost the average sign exponentially. In this formalism, we start with a trial density matrix and evolve it using a time-dependent Hamiltonian with a gradually increasing interaction strength. The adiabatic theorem guarantees that the evolved density matrix will approach the true ground-state provided the evolution time is long enough [30–33]. The main observation behind our method is that the average interaction strength is lowered in this method compared to the conventional QMC. It is empirically known that the aforementioned constant f is linearly proportional to the (average) interaction strength. Accordingly, the average sign will improve expo-

nentially in the AQMC approach.

The idea of adiabaticity can be applied to all different types of QMC, e.g., determinant QMC (DQMC) [8, 9], continuous time QMC [34, 35], constrained-path QMC [36], stochastic series expansions [37], etc. Furthermore, it is extremely straightforward to implement the AQMC by minimally modifying the available QMC codes. In the supplemental material (SM), we have provided a simple implementation of the AQMC algorithm in MATLAB which can be used to verify or extend our results.

Algorithm.— To elaborate on the details of the AQMC, let us study the nearest neighbor Hubbard model on the square lattice with the following Hamiltonian:

$$H_U = -t_1 \sum_{\langle \mathbf{ij} \rangle, \sigma} \hat{c}_{\mathbf{i},\sigma}^\dagger \hat{c}_{\mathbf{j},\sigma} - \mu \sum_{\mathbf{i},\sigma} \hat{n}_{\mathbf{i},\sigma}$$
$$+ U \sum_{\mathbf{i}} \left(\hat{n}_{\mathbf{i},\uparrow} - \frac{1}{2} \right) \left(\hat{n}_{\mathbf{i},\downarrow} - \frac{1}{2} \right), \qquad (1)$$

where $\langle \mathbf{ij} \rangle$ denotes the nearest neighbor sites, $\hat{c}_{\mathbf{i},\sigma}$ the annihilation operator of a spin σ electron at site i, and $\hat{n}_{\mathbf{i},\sigma} = \hat{c}_{\mathbf{i},\sigma}^\dagger \hat{c}_{\mathbf{i},\sigma}$ is the associated number operator. Throughout this letter, we choose $t_1 = 1$. The above model is proven to be sign-problem-free at half filling ($\mu = 0$) on bipartite lattices due to its time reversal (TR) and particle-hole (PH) symmetries [38]. However, finite doping $|\mu| > 0$ breaks the PH symmetry and introduces sign problem [39]. Motivated by the adiabatic theorem, we consider the following time-dependent auxiliary Hamiltonian:

$$H_{\mathscr{U}(\tau)} := -t_1 \sum_{\langle \mathbf{ij} \rangle, \sigma} \hat{c}_{\mathbf{i},\sigma}^\dagger \hat{c}_{\mathbf{j},\sigma} - \mu' \sum_{\mathbf{i},\sigma} \hat{n}_{\mathbf{i},\sigma}$$
$$+ \mathscr{U}(\tau) \sum_{\mathbf{i}} \left(\hat{n}_{\mathbf{i},\uparrow} - \frac{1}{2} \right) \left(\hat{n}_{\mathbf{i},\downarrow} - \frac{1}{2} \right), \qquad (2)$$

and a trial (initial) wave-function $|\Psi_T\rangle$ that has a non-vanishing overlap with the true ground-state of H_U, $|g\rangle_U$. The adiabatic theorem states that the following relation holds provided $H_{\mathscr{U}(\tau)}$ has a finite gap throughout the evolution (which

is always true for finite discrete systems), $\mathscr{U}(\tau)$ is a slowly varying function and $\lim_{\tau \to \beta/2} \mathscr{U}(\tau) = U$:

$$|g\rangle_U = \lim_{\beta \to \infty} \mathcal{T}_\tau e^{-\int_0^{\beta/2} H_{\mathscr{U}(\tau)} d\tau} |\Psi_T\rangle \qquad (3)$$

where, $\mathcal{T}_\tau$ stands for the (imaginary) time ordering. Since the total electron number commutes with $H_{\mathscr{U}}$ independent of $\mathscr{U}$, μ' (which enforces the electron density and is different from μ) can be time-independent. It is straightforward to reformulate the above relation using the density matrix formalism. Defining $\rho_U = |g\rangle_U \langle g|_U$, we have:

$$\rho_U = \lim_{\beta \to \infty} \mathcal{T}_\tau e^{-\int_0^{\beta/2} H_{\mathscr{U}(\tau)} d\tau} \rho_T \, \mathcal{T}_\tau e^{-\int_0^{\beta/2} H_{\overline{\mathscr{U}}(\tau)} d\tau}, \quad (4)$$

where, $\overline{\mathscr{U}}(\tau) = \mathscr{U}(\beta/2 - \tau)$, and ρ_T is a trial density matrix (where at least one of its nontrivial eigenstates has a non-vanishing overlap with $|g\rangle_U$). Obviously, a valid choice is $\rho_T = \mathbb{1}$.

The next step is to discretize the (imaginary) time axis and split the evolution time, $\beta/2$ into N steps, which leads to the following approximation of Eq. 4:

$$\rho_U \approx e^{-\Delta\tau H_{\mathscr{U}_N}} \cdots e^{-\Delta\tau H_{\mathscr{U}_1}} \rho_T e^{-\Delta\tau H_{\mathscr{U}_1}} \cdots e^{-\Delta\tau H_{\mathscr{U}_N}}, \quad (5)$$

in which $\Delta\tau = \frac{\beta}{2N}$, and $\mathscr{U}_l := \mathscr{U}(l\Delta\tau)$. After discretization, we employ the second-order Trotter-Suzuki decomposition to rewrite the exponential factors as:

$$e^{-\Delta\tau(H_K + H_{I,l})} = e^{-\frac{\Delta\tau}{2} H_K} e^{-\Delta\tau H_{I,l}} e^{-\frac{\Delta\tau}{2} H_K} + O(\Delta\tau^3). \quad (6)$$

where $H_{I,l} = \mathscr{U}_l \sum_{\mathbf{i}} \left(\hat{n}_{\mathbf{i},\uparrow} - \frac{1}{2} \right) \left(\hat{n}_{\mathbf{i},\downarrow} - \frac{1}{2} \right)$, and H_K represents the quadratic part of Eqs. 1 and 2. We then apply the Hirsch-Hubbard-Stratonovich transformation [7] to the Hubbard interaction after which:

$$e^{-\Delta\tau \mathscr{U}_l \left(\hat{n}_{\mathbf{i},\uparrow} - \frac{1}{2} \right) \left(\hat{n}_{\mathbf{i},\downarrow} - \frac{1}{2} \right)} = \frac{e^{-\frac{\Delta\tau}{4} \mathscr{U}_l}}{2} \sum_{\sigma = \pm} e^{\sigma \lambda_l (\hat{n}_{\mathbf{i},\uparrow} - \hat{n}_{\mathbf{i},\downarrow})}, \quad (7)$$

where $\cosh(\lambda_l) = e^{\Delta\tau \mathscr{U}_l / 2}$. The remaining steps are exactly identical to the regular DQMC [40].

In this letter, we make two different choices for the trial density matrix: (I) $\rho_T = \mathbb{1}$, and (II) $\rho_T = e^{-\beta_T H_K}$, i.e., a free fermion thermal ensemble at temperature $1/\beta_T$. Furthermore, we consider a bounded linear time dependence for the instantaneous onsite couplings: $\mathscr{U}(\tau) = \min \left(U, \mathscr{U}_{\max} \frac{2\tau}{\beta} \right)$. The convergence criteria for the AQMC is the flattening and convergence of energy and other desired observables against increasing β further.

Unlike a number of other powerful tools designed to mitigate the sign problem, e.g., constrained-path QMC [41], AQMC satisfies the variational theorem and therefore provides an upper bound for the ground-state energy. It is thus a controlled approximation in the sense that increasing the evolution/projection time, or optimizing the parameters of the trial density matrix will increase the accuracy of the method and will bring us closer to the true ground-state while never crossing it. Moreover, for symmetry preserving trial density

matrices, AQMC is an unbiased method and respects the ergodicity of the problem.

Benchmarking AQMC.— In the SM, we have benchmarked the AQMC algorithm by studying two cases with known exact results. We first compared its performance and accuracy with the regular DQMC for the Hubbard model on the square lattice at half filling, which is sign problem free, for a 16×2 square lattice subject to $U = 4$. As Fig. S6 of the SM shows, the AQMC recovers the exact results for sufficiently long evolution times (β). To further validate our method, we then considered the doped Hubbard model in the SM which is known to suffer from the sign problem. There, we studied a 16×4 square lattice on a cylinder, for $U = 4$ and at $p = 1/8$ doping . Fortunately, an accurate estimate of the ground-state energy for this problem is available from the DMRG approach. [42]. Again, our AQMC approach yields exact results for $\beta \gtrsim 10$ where the energy (per site) converges to $E = -1.0182 \pm 0.0003$ (see Fig. S7). Besides these two cases, in the SM, we have shown the superiority of our approach over the DQMC for other fillings, U values, lattice geometries and system sizes.

Results.— In this section, with the help of AQMC, we study two different strongly interacting models: (I) $p = 1/8$ doped Hubbard model on a periodic 16×8 square lattice and demonstrate the absence of superconductivity in the pure Hubbard model ($t' = 0$). (II) A highly interacting spin degenerate Chern insulator where we prove an emergent topological order identical to that of the bosonic $1/2$ Laughlin state.

I. $U = 4$ and $U = 8$ Hubbard model at $1/8$ doping on a periodic 16×8 system.— This filling has been the subject of intense research in the past few decades. There are numerous studies suggesting a plethora of various competing phases for the Hubbard model at $p = 1/8$ doping level with close energies including the d-wave superconductivity, and stripe order phases [42–53]. Our method can conveniently handle the system size and the boundary conditions considered here. Moreover, due to its enhanced average sign we can achieve the ground-state properties. This can be verified from Fig. 1b, which indicates that the average energy for $U = 4$ nearly plateaus around $E_g = -1.0305 \pm 0.0005$. The regular DQMC is obviously unable to reach beyond $\beta = 5.4$ due to its appalling sign problem (see Fig. 1a). On the other hand, DMRG is inapplicable for this system size and geometry. The torus boundary conditions aside, $N_y = 8$ requires gigantic bond dimensions for the convergence (as large as 10^6) which in currently unaffordable even on the best available computational facilities.

Our spin-spin correlation function clearly points towards the stripe order formation. As Fig. 1c shows, we evidence a π phase shift in the correlations after $|\Delta x| = 4$. This correlation function is consistent with a magnetization of the form $\langle S_z(x,y) \rangle = m \cos(Q_x x + Q_y y)$, where $(Q_x, Q_y) = \left(\frac{7}{8}, 1 \right) \pi$. We have also plotted the superconducting (pair-pair) correlation function $P_x(\mathbf{r}) = \langle \Delta_x(\mathbf{i})^\dagger \Delta_x(\mathbf{i} + \mathbf{r}) \rangle$ (averaged over all possible $\mathbf{i}$'s), where $\hat{\Delta}_x(\mathbf{i}) \equiv \hat{c}_{\mathbf{i},\uparrow} \hat{c}_{\mathbf{i}+\hat{x},\downarrow} - \hat{c}_{\mathbf{i},\downarrow} \hat{c}_{\mathbf{i}+\hat{x},\uparrow}$.
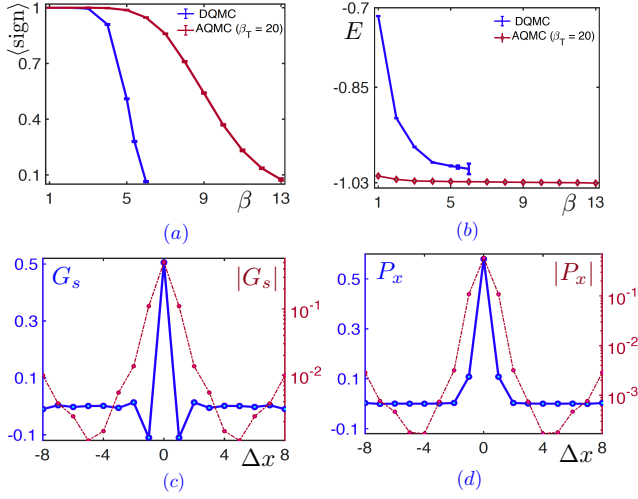
FIG. 1: (a)-(b) Comparing the average sign and energy of the regular DQMC and those of the AQMC ($\beta_T = 20$) against β for a 16×8 periodic system subject to $U = 4$ Hubbard interaction at $p = 1/8$ doping. (c)-(d) Magnetic and superconducting correlation functions along the x axis obtained via AQMC with $\beta = 12$, and $\beta_T = 20$ for the 16×8 sample at $p = 1/8$ and $U = 4$. The blue plots are in normal-normal scale and depict Green's functions and the red ones are in normal-log scale and represent the *absolute value* of the corresponding Green's functions. The spin-spin correlation function suggests a stripe order. In contrast, the pair-pair correlation function is relatively much smaller at long distances suggesting the absence of long-range superconducting order.

FIG. 2: AQMC U=8 results for an 16×8 system at $p = 1/8$ doping. ρ_T denotes the free Fermi surface trial density matrix, and ρ_T^* the optimized trial state with non-vanishing stripe order parameters. (a)-(b) Comparing the average sign and energy (versus β) of the AQMC with that of the DQMC methods. The AQMC can reach lower temperatures (β's) and yields lower energies. (c) The spin-spin and pair-pair correlation functions achieved via AQMC with ρ_T (blue) and ρ_T^* (orange) trial states. In both cases, the pair-pair correlation functions decay considerably faster at long distances, implying a non-superconducting state. Although ρ_T^* corresponds to a slightly lower energy state than ρ_T, they both consistently indicate the stripe order and belong to the same phase.

From the log-normal plots, it is clear that P_x is substantially more suppressed than G_s at long distances. Thus, we conclude that the ground-state is not a superconductor [40]. This is consistent with the growing agreement that for the pure Hubbard model ($t' = 0$), the stripe order wins the competition and the superconductivity is absent at this filling [54].

We now apply the AQMC to $U = 8$. As Fig. 2 implies, the AQMC outperforms the regular DQMC and achieves lower energies. Moreover, when we assume no ordering in ρ_T, and start with a free Fermi surface as our trial (initial) state, the energy versus β curve does not converge for $\langle \text{sign} \rangle > 0.1$. The lowest energy we obtain with this initial state is $E = -0.755 \pm 0.002$. For this state, similar to $U = 4$, the correlation functions point toward the stripe order phase. However, by modifying ρ_T to reflect the observed ordering (which we refer to as ρ_T^*), the convergence to the true ground-state can be achieved (with $E = -0.759 \pm 0.002$).

II. Correlated Chern insulators.—As our second nontrivial model Hamiltonian, we focus on a spin-degenerate staggered-flux Chern insulator [55] with strong onsite Hubbard attraction ($U = -6$) between electrons with opposite spins [40]. Here, the band-structure for spin up and down electrons contains two nearly flat bands with Chern number $C = \pm 1$ for
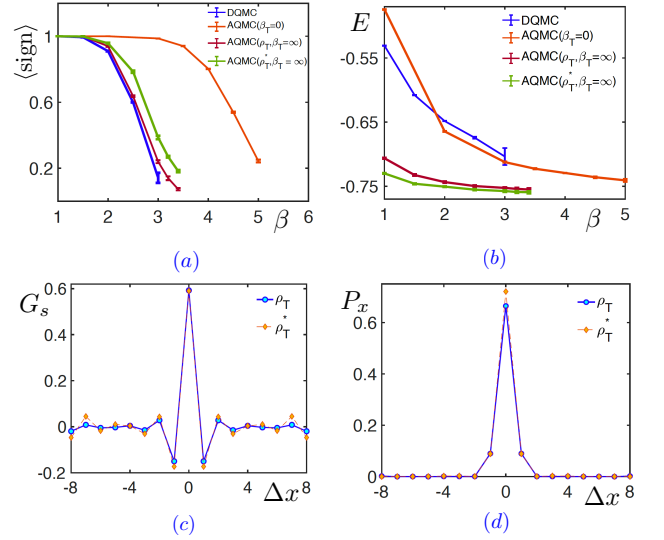
the valence and conduction bands, respectively. The system is at half filling, thus the valence band is fully occupied at $U = 0$. This model has a severe sign problem due to the lack of time-reversal symmetry [38, 40]. The fate of this model is not fully settled yet [56–62] although some evidences in favor of topological order have been found. There are several possible ground-states for this model for strongly interacting case including an s-wave superconductors [59, 62], a charge/spin density wave [61], and a nontrivial state with topological order similar to that of $\nu = 1/2$ Laughlin state [56–58]. In the latter phase, the fundamental degrees of freedom are charge $q = 2e$ doublons ($d_{\mathbf{r}} \equiv c_{\mathbf{r},\uparrow} c_{\mathbf{r},\downarrow}$), thus all spin-carrying operators such as electrons are confined [40]. Furthermore, the ground-state is two-fold degenerate on torus geometry, hosts non-trivial chiral edge states described by a $SU(2)_1 \equiv U(1)_2$ conformal field theory (CFT), and contains excitations with fractional charge ($q/2$) and fractional statistics ($\pi/2$) [56]. Our results below, exclude the former two candidates and supports the topological order. To carefully uncover the nature of the ground-state, we studied systems as large as 32×12 which only thanks to the AQMC became achievable.

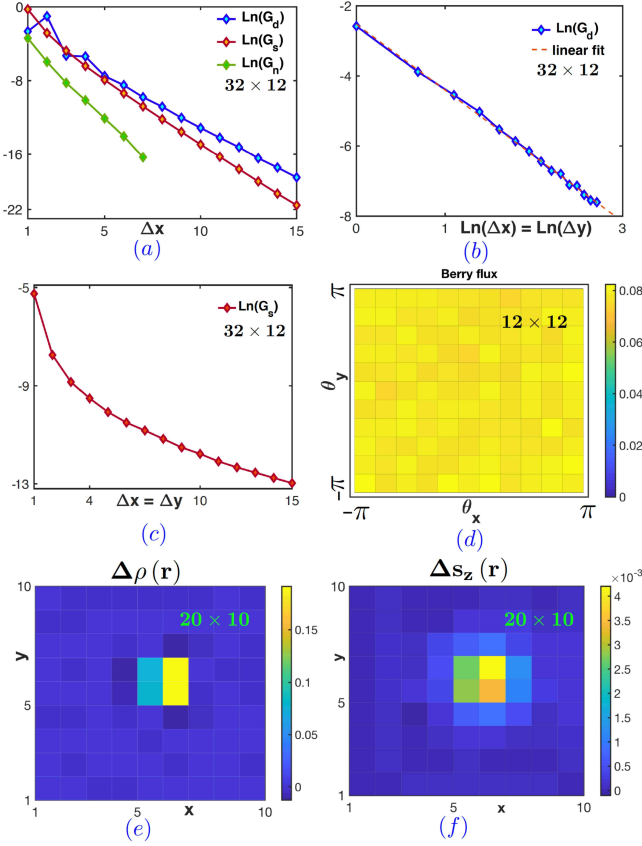We first exclude the possibility of s-wave superconductiv-

FIG. 3: The results of AQMC for a correlated spinful Chern insulator ($U = -6$). The system size for each plot is indicated. (a) The doublon-doublon, spin-spin, and density-density correlation functions die off exponentially in the bulk. The corresponding decay lengths are all less than a unit cell. (b) The doublon-doublon correlators decay algebraically as $1/r^{2h_d}$ at the edge of the system ($h_d \approx 0.95$), for an edge created along the $\hat{x} + \hat{y}$ direction. We obtain a nearly identical profile for the density-density correlation function consistent with an emergent $SU(2)$ edge symmetry. (c) Edge spin-spin correlators decay exponentially at long distances (with an enlarged decay length around 4 unit cells). (d) The many body Berry flux is nearly uniform in the entire phase space and its integral, the Berry phase, is fractional and equal to 0.98π in agreement with the theoretical predictions for a $\nu = 1/2$ Laughline state. (e) For a $\Phi_\uparrow = \Phi_\downarrow = hc/2e$, the total accumulated charge (relative to the background charge) equals 0.502 (in units of $2e$), while in (f) the accumulated spin in negligible ($s_{z,\text{tot}} = 0.02$) for $\Phi_\uparrow = -\Phi_\downarrow = hc/2e$. Note that only the left half space is shown and fluxes are inserted around the center.

ity and CDW order by studying the bulk pair-pair, density-density, and spin-spin correlation functions. As Fig. 3a suggests, all correlation functions decay exponentially in the bulk implying the absence of any symmetry breaking long range order. Next, we study the previously mentioned two-point correlators at the edge. For $U(1)_2 \equiv SU(2)_1$ CFT, we ex-

pect the density-density and pair-pair correlation functions to follow the same profile and both decay algebraically with $h_d = h_n = 1$ conformal dimensions, consistent with our results (see Fig. 3b). On the other hand, the topological order requires the spin degrees of freedom to be gapped even at the edge [40] and as a result its correlators must decay exponentially everywhere, consistent with our results in Fig. 3c. Another non-trivial fingerprint of the $1/2$ Laughline state is its fractional many body Chern number ($C_{\text{MB}} = 1/2$) [55]. The computation of C_{MB} involves applying twisted boundary conditions along x and y directions, and evaluating the overlap between many body wavefunctions with different twist angles [63]. As a result, similar to the single particle calculations, we can define the Berry curvature whose integral over the entire phase space determines the many body Chern number. In Fig. 3d we plot the Berry flux density which amounts to total Berry phase $\theta_B \approx \pi$. Hence $C_{\text{MB}} = \frac{\theta_B}{2\pi} \approx 1/2$.

Finally, theoretical considerations dictate that a quantum flux of doublons, i.e., $\Phi = hc/q = hc/2e$ will trap a nontrivial anyon excitation (known as semions in the $U(1)_2$ phase) with charge $q/2 = e$ bound to the flux location. In terms of the underlying microscopic degrees of freedom i.e., electrons, this translates into three different choices: (a) $\Phi_\uparrow(\mathbf{r}) = \Phi_\downarrow(\mathbf{r}) = hc/2e$, i.e., both electrons couple to the inserted fluxes symmetrically.(b) $\Phi_\uparrow = 2\Phi = hc/e$, and $\Phi_\downarrow = 0$. (c) $\Phi_\downarrow = 2\Phi = hc/e$, and $\Phi_\uparrow = 0$. In the $U(1)_2$ topological order spin degrees of freedom are gapped. Consequently, all the above three choices will lead to the same *spinless* charge e excitations, while in the non-interacting model, (a) is unacceptable since the fluxes are quantized in unites of hc/e, (b) will excite a spin up electron and (c) excites a spin-down electron. Therefore, the fractional excitations of $U(1)_2$ topological state (semions) are quite distinct from spinful electrons. Instead, they are *spinless fractionalized doublons* with *charge* $q/2 = e$. From the above discussion, we can conclude that in $U(1)_2$ topological order: (a) $\Phi_\uparrow = \Phi_\downarrow = hc/2e$ excites a semion with charge $q/2 = e$. (b) $\Phi_\uparrow = -\Phi_\downarrow = hc/2e$ is a trivial *spinless* and charge *neutral* excitation. Now, let us employ these two diagnostics. In Fig. 3e, we considered $\Phi_\uparrow(\mathbf{r}) = \Phi_\downarrow(\mathbf{r}) = hc/2e$ flux insertions. As we see, the total excess charge is quantized at $q/2 = e$ and localized around the inserted flux. More importantly, in Fig. 3f, we considered $\Phi_\uparrow(\mathbf{r}) = -\Phi_\downarrow(\mathbf{r}) = hc/2e$ flux insertion. We observe that this corresponds to a neutral excitation with a negligible accumulated spin, consistent with theory.

Discussion.—So far, we considered the simplest form of the AQMC algorithm. Our trial density matrix was simply related to the hopping Hamiltonian as $\rho_T = e^{-\beta_T H_K}$. We can in principle consider more general choices for ρ_T (ρ_T^*), and replace H_K by a variational H_K^* with variational inputs, e.g., order parameters of relevant symmetry breaking phases and optimize the ground-state energy with respect to them. Another direction to improve AQMC is to consider an interacting trial density matrix by utilizing the variational QMC algorithm [26]. In this approach, we can reach the true ground-state with shorter evolutions. Finally, the profile of $\mathcal{U}(\tau)$ can

be treated as a variational function. Although we applied the AQMC to the Hubbard model within the DQMC framework, it is not limited to them. We can apply it to other fermionic or bosonic model Hamiltonians or implement it within other QMC frameworks. Among those model Hamiltonians, in particular we would like to mention the multi-layer Hubbard model and the partially flat band systems [64–66]. The average sign is already larger in these models and can be boosted further using AQMC such that the ground-state properties will be more accurately captured.

[1] R. Blankenbecler, D. Scalapino, and R. Sugar, Physical Review D **24**, 2278 (1981).
[2] S. R. White, D. J. Scalapino, R. L. Sugar, E. Y. Loh, J. E. Gubernatis, and R. T. Scalettar, Phys. Rev. B **40**, 506 (1989).
[3] W. von der Linden, Physics Reports **220**, 53 (1992).
[4] D. Leinweber, (2000).
[5] W. Foulkes, L. Mitas, R. Needs, and G. Rajagopal, Reviews of Modern Physics **73**, 33 (2001).
[6] A. Gezerlis, I. Tews, E. Epelbaum, S. Gandolfi, K. Hebeler, A. Nogga, and A. Schwenk, Physical review letters **111**, 032501 (2013).
[7] J. E. Hirsch, Physical Review B **28**, 4059 (1983).
[8] F. F. Assaad, Quantum Simulations of Complex Many-Body Systems: From Theory to Algorithms **10**, 99 (2002).
[9] R. R. d. Santos, Brazilian Journal of Physics **33**, 36 (2003).
[10] M. Troyer and U.-J. Wiese, Physical review letters **94**, 170201 (2005).
[11] E. Berg, M. A. Metlitski, and S. Sachdev, Science **338**, 1606 (2012).
[12] Z.-X. Li, A. Vaezi, C. B. Mendl, and H. Yao, Science advances **4**, eaau1463 (2018).
[13] E. Berg, S. Lederer, Y. Schattner, and S. Trebst, Annual Review of Condensed Matter Physics **10**, 63 (2019).
[14] C. Varney, C.-R. Lee, Z. Bai, S. Chiesa, M. Jarrell, and R. Scalettar, Physical Review B **80**, 075116 (2009).
[15] F. F. Assaad and I. F. Herbut, Physical Review X **3**, 031010 (2013).
[16] T. Paiva, R. R. Dos Santos, R. Scalettar, and P. Denteneer, Physical Review B **69**, 184501 (2004).
[17] Z.-X. Li, Y.-F. Jiang, and H. Yao, Physical Review B **91**, 241117 (2015).
[18] Z.-X. Li and H. Yao, Annual Review of Condensed Matter Physics **10**, 337 (2019).
[19] Z.-X. Li, Y.-F. Jiang, and H. Yao, Phys. Rev. Lett. **117**, 267002 (2016).
[20] L. Wang, Y.-H. Liu, M. Iazzi, M. Troyer, and G. Harcos, Physical review letters **115**, 250601 (2015).
[21] Z. Wei, C. Wu, Y. Li, S. Zhang, and T. Xiang, Physical review letters **116**, 250601 (2016).
[22] E. Huffman and S. Chandrasekharan, Physical Review D **96**, 114502 (2017).
[23] F. Alet, K. Damle, and S. Pujari, Physical review letters **117**, 197203 (2016).
[24] E. Loh Jr, J. Gubernatis, R. Scalettar, S. White, D. Scalapino, and R. Sugar, Physical Review B **41**, 9301 (1990).
[25] C. Umrigar, J. Toulouse, C. Filippi, S. Sorella, and R. G. Hennig, Physical review letters **98**, 110201 (2007).
[26] M.-S. Vaezi and A. Vaezi, arXiv preprint arXiv:1810.00864 (2018).
[27] Z.-Q. Wan, S.-X. Zhang, and H. Yao, arXiv preprint arXiv:2010.01141 (2020).
[28] D. Hangleiter, I. Roth, D. Nagaj, and J. Eisert, Science advances **6**, eabb8341 (2020).
[29] R. Levy and B. K. Clark, arXiv preprint arXiv:1907.02076 (2019).
[30] G. Nenciu, Journal of Physics A: Mathematical and General **13**, L15 (1980).
[31] Z. Wu and H. Yang, Physical Review A **72**, 012114 (2005).
[32] J. E. Avron and A. Elgart, Communications in mathematical physics **203**, 445 (1999).
[33] C.-W. Liu, A. Polkovnikov, and A. W. Sandvik, Physical Review B **87**, 174302 (2013).
[34] A. N. Rubtsov, V. V. Savkin, and A. I. Lichtenstein, Physical Review B **72**, 035122 (2005).
[35] E. Gull, A. J. Millis, A. I. Lichtenstein, A. N. Rubtsov, M. Troyer, and P. Werner, Reviews of Modern Physics **83**, 349 (2011).
[36] S. Zhang, J. Carlson, and J. E. Gubernatis, Physical Review B **55**, 7464 (1997).
[37] A. W. Sandvik, Physical Review B **59**, R14157 (1999).
[38] C. Wu and S.-C. Zhang, Physical Review B **71**, 155115 (2005).
[39] V. Iglovikov, E. Khatami, and R. Scalettar, Physical Review B **92**, 045110 (2015).
[40] A. For more details and additional information see the Supplemental Material .
[41] J. Carlson, J. Gubernatis, G. Ortiz, and S. Zhang, Physical Review B **59**, 12788 (1999).
[42] G. Ehlers, S. White, and R. Noack, Physical Review B **95**, 125125 (2017).
[43] S. A. Kivelson, I. P. Bindloss, E. Fradkin, V. Oganesyan, J. Tranquada, A. Kapitulnik, and C. Howald, Reviews of Modern Physics **75**, 1201 (2003).
[44] S.-X. Yang, H. Fotso, S.-Q. Su, D. Galanakis, E. Khatami, J.-H. She, J. Moreno, J. Zaanen, and M. Jarrell, Physical review letters **106**, 047004 (2011).
[45] E. W. Huang, C. B. Mendl, S. Liu, S. Johnston, H.-C. Jiang, B. Moritz, and T. P. Devereaux, Science **358**, 1161 (2017).
[46] M. Dolfi, B. Bauer, S. Keller, and M. Troyer, Physical Review B **92**, 195139 (2015).
[47] H.-C. Jiang and T. P. Devereaux, Science **365**, 1424 (2019).
[48] J. F. Dodaro, H.-C. Jiang, and S. A. Kivelson, Physical Review B **95**, 155116 (2017).
[49] A. S. Darmawan, Y. Nomura, Y. Yamaji, and M. Imada, Physical Review B **98**, 205132 (2018).
[50] B.-X. Zheng and G. K.-L. Chan, Physical Review B **93**, 035126 (2016).
[51] B.-X. Zheng, C.-M. Chung, P. Corboz, G. Ehlers, M.-P. Qin, R. M. Noack, H. Shi, S. R. White, S. Zhang, and G. K.-L. Chan, Science **358**, 1155 (2017).

[52] J. P. LeBlanc, A. E. Antipov, F. Becca, I. W. Bulik, G. K.-L. Chan, C.-M. Chung, Y. Deng, M. Ferrero, T. M. Henderson, C. A. Jiménez-Hoyos, *et al.*, Physical Review X **5**, 041041 (2015).

[53] P. Corboz, T. M. Rice, and M. Troyer, Physical review letters **113**, 046402 (2014).

[54] M. Qin, C.-M. Chung, H. Shi, E. Vitali, C. Hubig, U. Schollwöck, S. R. White, S. Zhang, *et al.*, Physical Review X **10**, 031016 (2020).

[55] T. Neupert, L. Santos, C. Chamon, and C. Mudry, Physical review letters **106**, 236804 (2011).

[56] J. Maciejko and A. Rüegg, Physical Review B **88**, 241101 (2013).

[57] A. Rüegg and G. A. Fiete, Physical review letters **108**, 046401 (2012).

[58] J. He, S.-P. Kou, Y. Liang, and S. Feng, Physical Review B **83**, 205116 (2011).

[59] C. Hickey, P. Rath, and A. Paramekanti, Physical Review B **91**, 134414 (2015).

[60] J. Wu, J. P. L. Faye, D. Sénéchal, and J. Maciejko, Physical Review B **93**, 075131 (2016).

[61] V. Arun, R. Sohal, C. Hickey, and A. Paramekanti, Physical Review B **93**, 115110 (2016).

[62] Y.-C. Zhang, Z. Xu, and S. Zhang, Physical Review A **95**, 043640 (2017).

[63] A. Moharramipour, et al., in preparation.

[64] E. W. Huang, M.-S. Vaezi, Z. Nussinov, and A. Vaezi, Physical Review B **99**, 235128 (2019).

[65] S. Sayyad, E. W. Huang, M. Kitatani, M.-S. Vaezi, Z. Nussinov, A. Vaezi, and H. Aoki, Physical Review B **101**, 014501 (2020).

[66] H. Aoki, Journal of Superconductivity and Novel Magnetism , 1 (2020).

SUPPLEMENTAL MATERIAL

In this supplemental material (SM), we first delve into the details of the adiabatic quantum Monte Carlo (AQMC) algorithm. Next, we compare the energy and average sign of the AQMC with those of the standard determinant quantum Monte Carlo (DQMC) for various system sizes, interaction strengths, doping levels, and two different lattice geometries. In section 3 of this SM, we benchmark the AQMC method for two cases with known exact results to prove its accuracy and reliability. In section 4, we present more results for the Hubbard model at $p = 1/8$. In section 5, we argue that strong attractions may lead to the emergence of a $\nu = 1/2$ Laughline state from correlated Chern insulator. At the end, in section 6, we provide a simple implementation of the determinant and adiabatic quantum Monte Carlo in a single script in the MAT-LAB environment.

1. DETAILS OF THE AQMC ALGORITHMS

Here we discuss the implementation of the AQMC algorithm in more detail. It involves very similar computational steps to a regular determinant quantum Monte Carlo (DQMC).

As it is discussed in the main text, the following expression considered as an approximation to the true density matrix (and becomes exact in the $\beta \to \infty$ limit):

$$\rho_U \approx e^{-\Delta\tau H_{\mathscr{U}_N}} \cdots e^{-\Delta\tau H_{\mathscr{U}_1}} \rho_T e^{-\Delta\tau H_{\mathscr{U}_1}} \cdots e^{-\Delta\tau H_{\mathscr{U}_N}}. \tag{S1}$$

The expectation value of any physical operator (e.g., the Green's function) is obtained via the following relation:

$$\langle G \rangle = \frac{\text{Tr}(G\rho_U)}{\text{Tr}(\rho_U)}. \tag{S2}$$

After performing the Hubbard-Stratonovich (HS) transformation, it can be shown that the Green's function for spin σ electrons can be written as

$$\langle G^\sigma \rangle = \frac{\sum_{\mathbf{s}} G^\sigma_{\mathbf{s}} \mathcal{W}_{\mathbf{s}}}{\sum_{\mathbf{s}} \mathcal{W}_{\mathbf{s}}} = \frac{\langle G^\sigma_{\mathbf{s}} \text{sign}(\mathcal{W}_{\mathbf{s}}) \rangle}{\langle \text{sign}(\mathcal{W}_{\mathbf{s}}) \rangle}, \tag{S3}$$

where the expectation values over the Hubbard-Stratonovich field can be computed using Metropolis-Hastings sampling algorithm with the acceptance probability of $P(\mathbf{s} \to \bar{\mathbf{s}}) = \min(\frac{|\mathcal{W}_{\bar{\mathbf{s}}}|}{|\mathcal{W}_{\mathbf{s}}|}, 1)$ and $G^\sigma_{\mathbf{s}}$ and $\mathcal{W}_{\mathbf{s}}$ are given by

$$G^\sigma_{\mathbf{s}} = (I + B^\sigma_{2N+N'} \cdots B^\sigma_{N+1} B^\sigma_N \cdots B^\sigma_1)^{-1}, \tag{S4}$$

$$\mathcal{W}_{\mathbf{s}} = (\det G^\uparrow_{\mathbf{s}})^{-1} (\det G^\downarrow_{\mathbf{s}})^{-1}, \tag{S5}$$

where the B matrices are defined as

$$B^\sigma_l = e^{-\Delta\tau \mathsf{K}} e^{-dig(\lambda_l \sigma s_i(l))}, \tag{S6}$$

with

$$K_{ij} = \begin{cases} -t & \text{if } i \text{ and } j \text{ are nearest neighbors,} \\ -\mu' & \text{if } i = j, \\ 0 & \text{otherwise} \end{cases}, \tag{S7}$$

$$\mathscr{U}_l = \begin{cases} \frac{U}{N}(N - l) & \text{if } l \leq N, \\ 0 & \text{if } N < l < N + N', \\ \frac{U}{N}(l - (N + N')) & \text{if } N + N' \leq l \end{cases} \tag{S8}$$

and $\cosh \lambda_l = e^{\Delta\tau \mathscr{U}_l / 2}$. The correlation functions other than the fermionic Green's function can be also found using the Wick's theorem which is valid for each Monte Carlo sampling step.

In order to update Monte Carlo configuration and walk through the Markov chain, we have to check whether the HS spin at each space-time point would be flipped or not. To perform this update, we use the Sherman-Morrison update rule which is more efficient than calculating the acceptance probability and the updated Green's function from scratch and only requires $\mathcal{O}(N_s^2)$ operations rather than $\mathcal{O}(N_s^3)$ which is the typical computational complexity of matrix manipulations.

Using Sherman-Morrison formulas, one can prove that the following relations can be applied to find the acceptance ratio and updated Green's function with HS spin at time $l = 1$ and site i flipped.

$$\alpha^\sigma(i) \equiv e^{-2\lambda_l \sigma s_i} - 1, \tag{S9}$$

$$r^\sigma(i) = \frac{\det(\bar{G}^\sigma)^{-1}}{\det(G^\sigma)^{-1}} = 1 + \alpha^\sigma(i)(1 - G^\sigma_{ii}), \tag{S10}$$

$$\bar{G}^\sigma_{jk} = G^\sigma_{jk} - \frac{\alpha^\sigma(i)}{r^\sigma(i)} \left[\delta_{ji} - G^\sigma_{ji}\right] G^\sigma_{ik} \tag{S11}$$

To update HS spins at time $l = 2$, we need to bring the B_2 matrix the rightmost place of $B^\sigma_{2N+N'} \cdots B^\sigma_{N+1} B^\sigma_N \cdots B^\sigma_1$ This can be easily done by the following transformation,

$$G^\sigma(l + 1) = B^\sigma_l G^\sigma(l)[B^\sigma_l]^{-1} \tag{S12}$$

Since the wrap up operation does not change the determinant, Sherman-Morrison updating formulas are still allowed. By repeating this "update-wrap up" process, one can sweep the imaginary time direction and update all of the HS spins. After each wrap up, the $\mathscr{U}_l$ gets shifted circularly (see Fig. (4)) and finally after a complete space-time sweep returns to its initial form. Hence, physical measurements have to be done once at the end of each space-time sweep where the $\mathscr{U}_l$ profile is symmetric around $L/2$ ($L = 2N + N'$ is the total number of time steps).

Furthermore, in order to decrease the Trotter error that arises due to time discretization, one could employ second order Trotter-Suzuki decomposition which is
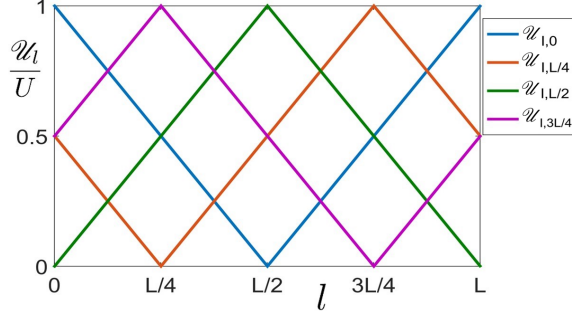
FIG. 4: Plots of $\mathscr{U}_{l,l_0}$ for some different values of l_0 which indicates the number of time slice shifts. For the sake of clarity, here we have considered $\beta_T = 0$ ($N' = 0$).

$$e^{-\Delta\tau K - \Delta\tau V} = e^{-\frac{\Delta\tau K}{2}} e^{-\Delta\tau V} e^{-\frac{\Delta\tau K}{2}} + \mathcal{O}(\Delta\tau^3) \quad \text{(S13)}$$

and leads to an error of $\mathcal{O}(\Delta\tau^3)$ rather than $\mathcal{O}(\Delta\tau^2)$. Since, G^σ is determined in terms of the product of $\mathcal{O}(\frac{1}{\Delta\tau})$ B matrices, its Trotter error would be $\mathcal{O}(\Delta\tau^2)$ in this case. More specifically, the second order Trotter error is $\mathcal{O}\left(\frac{\Delta\tau^3}{24}\left(2[V,[K,V]] + [K,[K,V]]\right)\right)$. Since the interaction strength is about several times greater than the hopping amplitude ($t = 1$ in our code), the latter commutator is negligible compared to the former. Therefore, the Trotter error would be proportional to $\mathscr{U}(\tau)^2 \Delta\tau^3$ at a single (imaginary) time step. Basically, the overall error is obtained by integrating the errors at single time steps over time. For the AQMC case in which $\mathscr{U}(\tau)$ is a linear function, this error would be proportional to $\mathcal{O}(\beta U^2 \Delta\tau^2/3)$ which is three times smaller than the error for the regular DQMC ($\mathscr{U}(\tau) = \text{const.}$).

According to the Eq. (S4) which determines G^σ in terms of B matrices, it is easy to show that it is sufficient to modify the Green's function by $G^\sigma_{\text{2nd order}} = e^{-\frac{\Delta\tau K}{2}} G^\sigma_{\text{1st order}} e^{\frac{\Delta\tau K}{2}} = e^{-\frac{\Delta\tau K}{2}} G^\sigma e^{\frac{\Delta\tau K}{2}}$ in order to use the second order Trotter-Suzuki decomposition for measuring physical quantities after each space-time sweep.

2. A SYSTEMATIC COMPARISON BETWEEN AQMC AND DQMC

Below, in Fig. 5 we have applied the AQMC algorithm to $U = 4, 6, 8$, and $p = 1/8$ for $N_x = N_y = 8$ square geometry. As we see, the AQMC ameliorates the sign problem for all three cases, and achieves a lower energy than the standard DQMC.

In Fig. 6, we have considered the following three different system sizes: 4×4, 8×8, and 12×12. In every case, we conclude that it can access a β larger the maximum β available via the standard DQMC, results an exponentially larger average sign in the same β, and above all yields a lower en-
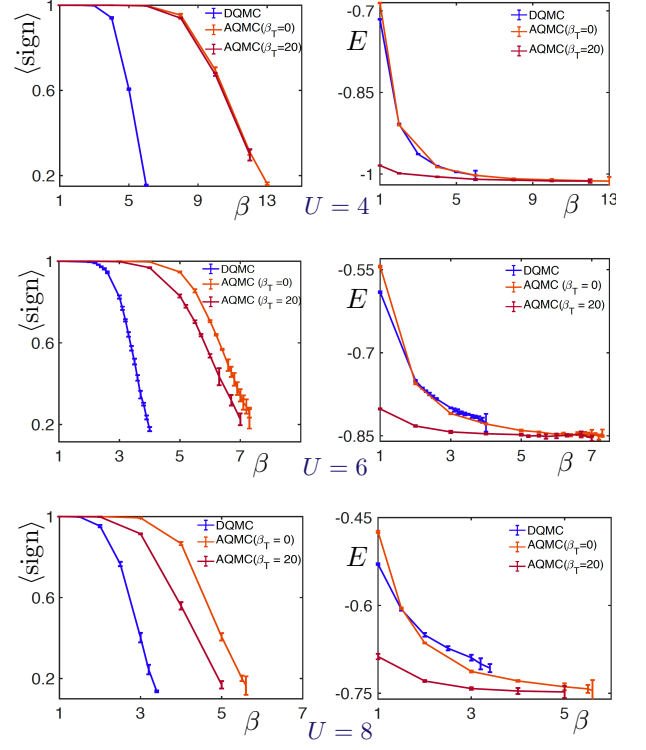
FIG. 5: Comparing the AQMC with DQMC for $U = 4, 6, 8$ at $p = 1/8$ doping ($< n >= 0.875$) for an 8×8 square geometry.

ergy estimate and therefore has a larger overlap with the true ground-state.

In Fig. 7, we have considered the following three doping levels: $p = 0.2, 1/8, 0.05$ corresponding to $< n >= 0.8, 0.875, 0.95$ fillings, respectively. Again, for all fillings our expectations are fulfilled.

In Fig. 8, we show that the triangular model which has geometric frustration can be studied by our approach while in the standard DQMC, it is not possible below $T = 1/3$ (beyond $\beta = 3$).

3. BENCHMARKING AQMC

In this section, we benchmark the AQMC algorithm by studying two cases with known exact results. First, let us compare its performance and accuracy with the regular DQMC for the Hubbard model on the square lattice at half filling. Since this situation does not suffer from the sign problem, we can consider arbitrarily low temperatures and achieve the true ground-state properties within the regular DQMC. To this end, we consider a two-leg ladder with $(N_x, N_y) = (16, 2)$ dimensions where electrons are coupled through $U = 4$. We define

FIG. 6: Comparing the AQMC with DQMC for $U = 4$ at $p = 1/8$ doping for 4×4, 8×8, and 12×12 size.

the average energy per site as

$$E = \frac{1}{N_s} \left(U \sum_{\mathbf{i}} \langle \hat{n}_{\mathbf{i},\uparrow} \hat{n}_{\mathbf{i},\downarrow} \rangle - t_1 \sum_{\langle \mathbf{ij} \rangle, \sigma} \langle \hat{c}^{\dagger}_{\mathbf{i},\sigma} \hat{c}_{\mathbf{j},\sigma} \rangle \right), \quad (8)$$

where $N_s = N_x N_y$ denotes the number of sites. In Fig. 9a, we have plotted the average energy at temperature $T = 1/\beta$ for the DQMC for several temperatures down to $T = 1/40$ (in units of t_1). We then consider the same β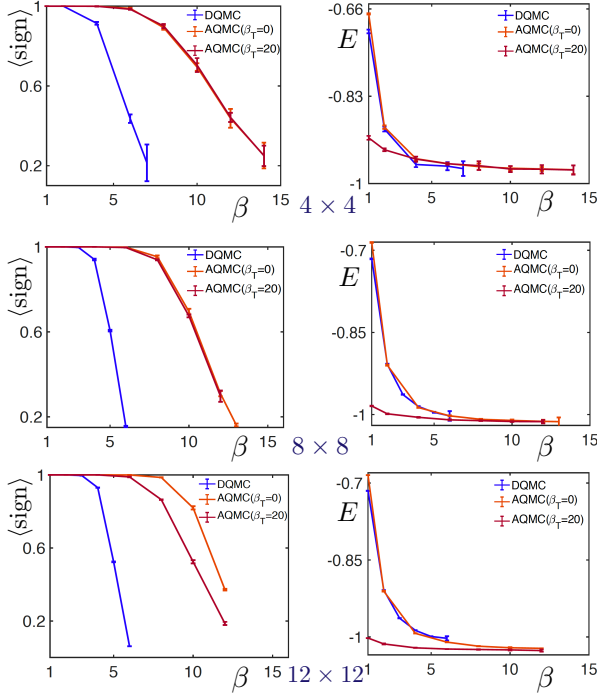 for the AQMC first for $\beta_T = 0$ and then for $\beta_T = 20$. As Fig. 9 clearly indicates, all three methods yield the same ground-state energy per site for large values of β within the statistical error bars. For DQMC, we obtain $E = -0.7086 \pm 0.0003(-0.7091 \pm 0.0003)$ at $\beta = 20$ $(\beta = 40)$. For $\beta_T = 0$, we achieve $E = -0.7085 \pm 0.0003$ for $\beta = 20$ and -0.7088 ± 0.0003 for $\beta = 40$. On the other hand for $\beta_T = 20$, we obtain $E = -0.7092 \pm 0.0003$ for both $\beta = 20$ and $\beta = 40$. Indeed, $\beta_T = 20$ is closely related to the projector QMC technique which is known to reach the ground-state for lower values of β (compared to the regular DQMC). In Fig. 9, besides the average energy, we have also compared the single-particle Green's function (for spin-up electrons), $G_f(\mathbf{r}) = \langle c^{\dagger}_{\mathbf{i},\uparrow} c_{\mathbf{i}+\mathbf{r},\uparrow} \rangle$, spin-spin correlation function, $G_s(\mathbf{r}) = \langle \mathbf{s_i} . \mathbf{s_{i+r}} \rangle$, and the pair distribution function for spin-up electrons, $G_n(|\mathbf{r}| > 0) = \langle n_{\mathbf{i},\uparrow} n_{\mathbf{i}+\mathbf{r},\uparrow} \rangle$, and for $\mathbf{r} = \Delta x \hat{x}$. The results are fairly consistent.

To further validate our method, we now focus on the doped
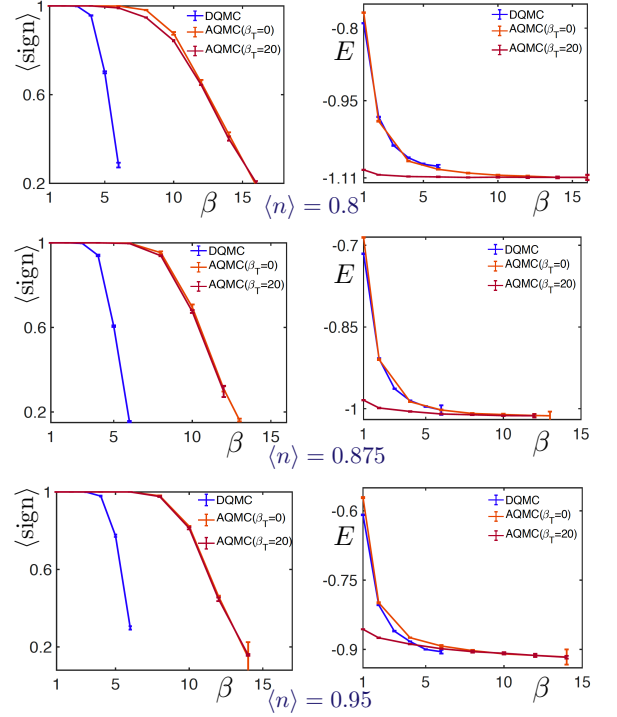
FIG. 7: Comparing the AQMC with DQMC for $< n > = 0.8, 0.875, 0.9$ at $U = 4$ for an 8×8 square geometry.
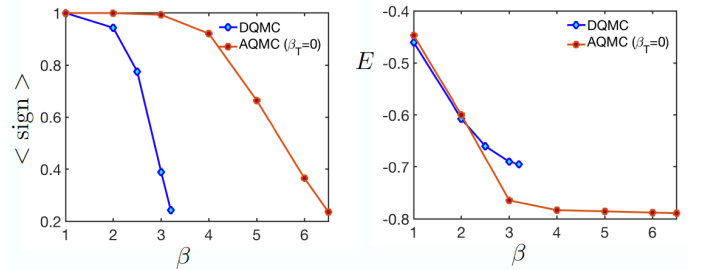
FIG. 8: Comparing the AQMC with DQMC for the Hubbard model above the Mott transition ($U = 6$) at half filling on an 8×8 triangular geometry.

Hubbard model which is known to suffer from the sign problem. Here, we consider $(N_x, N_y) = (16, 4)$, $U = 4$, and $p = 1/8$ on a cylinder geometry. Fortunately, an accurate estimate of the ground-state energy for this problem is available from the DMRG approach. In Ref. [42], this and a number of other related problems have been studied by considering bond dimensions as large as $M = 35000$, and their finite M results are then extrapolated to $M = \infty$ (more precisely the zero truncation error limit) to extract the true ground-state energy. Although for this particular case, the authors have only reported their infinite M extrapolation results with $E_g = -1.01841$ for the ground-state energy per site, from their other available extrapolations, we believe that for 16×4 system the average truncation error is about 10^{-5} for $M =$

FIG. 9: Comparing the AQMC and DQMC for a 16×2 system with $U = 4$ at half filling. (a) Average energy obtained via DQMC, AQMC ($\beta_T = 0$) and AQMC($\beta_T = 20$) for several values of β. (b)-(d) Pair-wise comparison for the Green's function, spin-spin correlation function and pair distribution functions along the x direction at $\beta = 40$.
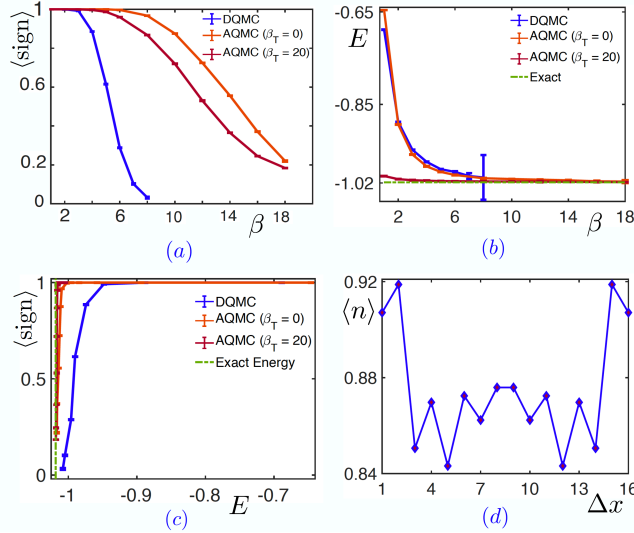
FIG. 10: (a)-(c) Comparing the results of AQMC with those obtained via DQMC and the exact result from DMRG's zero-truncation-error extrapolation for a 16×4 cylindrical system at $p = 1/8$ doping with $U = 4$. (d) Local electron density achieved from AQMC with $\beta = 18$, and $\beta_T = 20$.

35k. Therefore DMRG's actual estimate at this bond dimension is indeed approximately $E = -1.0175 \pm 0.0005$. Note that such an enormous bond dimension will require weeks of CPU-time on state-of-the-art supercomputers. In contrast, we achieved $E = -1.0182 \pm 0.0003$ using the AQMC (for $\beta = 18$, $\beta_T = 20$) within a few hours on a personal computer.

In Fig. 10a, we have plotted the average sign at a given β

FIG. 11: Various correlation functions along the x axis obtained via AQMC with $\beta = 12$, and $\beta_T = 20$ for the 16×8 sample at $p = 1/8$ and $U = 4$: (a) Green's function of spin up ele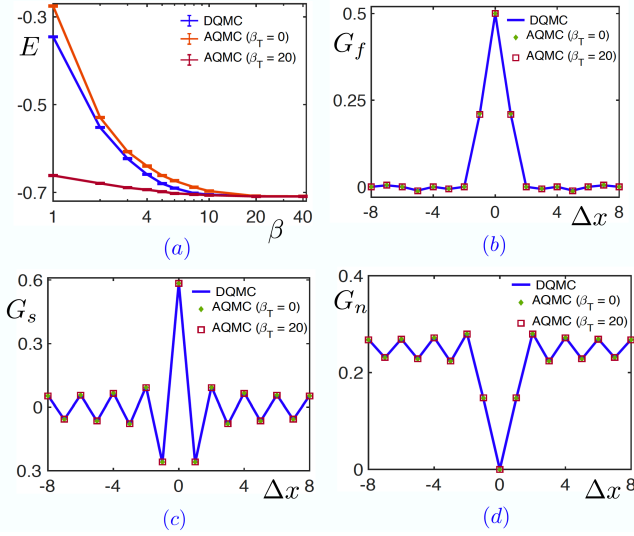ctrons, (b) Spin-spin correlations function, (c) and (d) Superconducting (pair-pair) correlation function for x and y bonds, respectively.

for the regular DQMC, as well as the AQMC for both $\beta_T = 0$, and $\beta_T = 20$. As promised earlier, the average sign is significantly higher in the AQMC. In Fig. 10b, the average energy as a function of β is presented for all three simulations. Again, we can verify that the AQMC, especially for $\beta_T = 20$, almost reaches the true ground-state energy (corroborated via DMRG results extrapolated to $M = \infty$ [42]) within the statistical error bar of our samplings. On the other hand, due to the severity of the sign problem, for the regular DQMC, we could not draw any meaningful conclusion beyond $\beta = 7$ where $\langle \text{sign} \rangle \approx 0.1$ and $E \approx -1.007 \pm 0.005$. Another useful plot is the average sign as a function of average energy. In the AQMC, we can probe ground-state properties before the sign problem becomes uncontrollable. Finally, we have plotted the electron density as a function of x which is similar to Fig. 6a of Ref. [42] (though they have plotted it for a 32×4 cylinder).

4. MORE CORRELATION FUNCTIONS FOR 16×8 SYSTEM

We here provide more details about the $U = 4$, 16×8 system at $p = 1/8$ doping level. For the two-point correlations, we have considered $\beta = 12$, $\beta_T = 20$ with about seven million spacetime sweeps in total. We have measured the spin-resolved Green's function ($G_f (\mathbf{r})$), spin-spin ($G_s (\mathbf{r})$) and also the superconducting (pair-pair) correlation functions ($P_x (\mathbf{r})$ and $P_y (\mathbf{r})$). The pair-pair correlations are related to the following two pairing fields: $\hat{\Delta}_x (\mathbf{i}) \equiv \hat{c}_{\mathbf{i},\uparrow} \hat{c}_{\mathbf{i}+\hat{x},\downarrow} - \hat{c}_{\mathbf{i},\downarrow} \hat{c}_{\mathbf{i}+\hat{x},\uparrow}$ and $\hat{\Delta}_y (\mathbf{i}) \equiv \hat{c}_{\mathbf{i},\uparrow} \hat{c}_{\mathbf{i}+\hat{y},\downarrow} - \hat{c}_{\mathbf{i},\downarrow} \hat{c}_{\mathbf{i}+\hat{y},\uparrow}$. Accord-
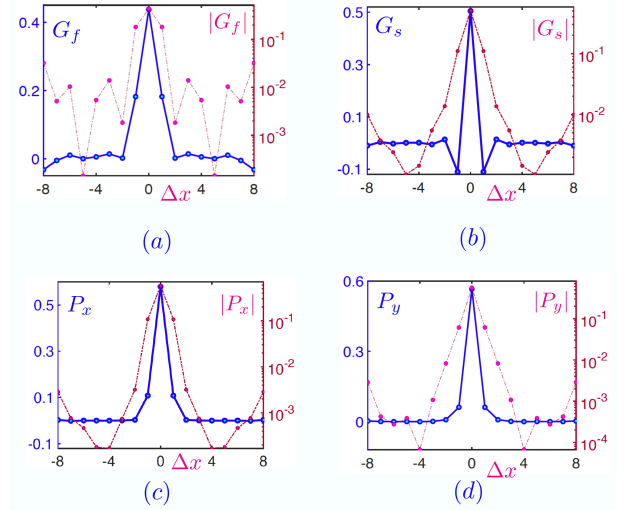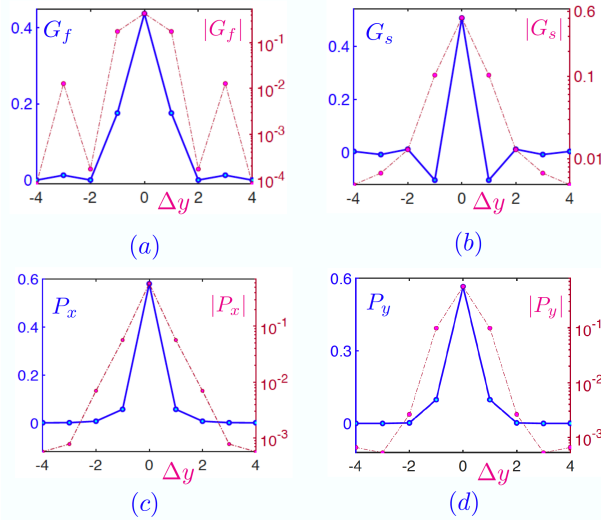
FIG. 12: Various correlation functions along the y axis obtained via AQMC with $\beta = 12$, and $\beta_T = 20$ for the 16×8 sample at $p = 1/8$ and $U = 4$: (a) Green's function of spin up electrons, (b) Spin-spin correlations function, (c) and (d) Superconducting (pair-pair) correlation function for x and y bonds, respectively.

ingly, we define $P_x(\mathbf{r}) = \langle \Delta_x(\mathbf{i})^\dagger \Delta_x(\mathbf{i} + \mathbf{r}) \rangle$, $P_y(\mathbf{r}) = \langle \Delta_y(\mathbf{i})^\dagger \Delta_y(\mathbf{i} + \mathbf{r}) \rangle$. We employ the translation symmetry and average over all possible $\mathbf{i}$'s. Figs. 11 and 12 summarize our results.

Our spin-spin correlation function clearly points towards the stripe order formation. As Fig. 11b shows, we evidence a π phase shift in the correlations after $|\Delta x| = 4$. While for $|\Delta x| \leq 4$, spins with odd (even) distances along the x axis (and with identical y's) have negative (positive) correlations, we observe the opposite behavior for $4 < |\Delta x| \leq 8$. Another exotic feature of the correlations is that they are all enhanced near $|\Delta x| = 8$, and grow with distance (rather than decay) for $5 < |\Delta x| \leq 8$. In contrast, correlation functions along the y axis (c.f., Fig. 12) behave normally and as expected.

5. EMERGENCE OF $\nu = 1/2$ LAUGHLINE STATE IN THE CORRELATED CHERN INSULATORS

Now, we would like to elucidate how the topological order can emerge from interacting Chern insulators. Our approach is closely related to the parton construction of fractional quantum Hall states. First, let us recall that the low energy description of a non-interacting Chern insulator is given by a Chern-Simons gauge theory after integrating out the fermionic degrees of freedom. Thus, we have:

$$\mathcal{L}_{\text{tot}} = \mathcal{L}_\uparrow + \mathcal{L}_\downarrow,$$
$$\mathcal{L}_{\sigma = \uparrow, \downarrow} = \frac{1}{4\pi} \epsilon^{\mu\nu\lambda} a_{\mu,\sigma} \partial_\nu a_{\lambda,\sigma} - \frac{e}{2\pi} \epsilon^{\mu\nu\lambda} a_{\mu,\sigma} \partial_\nu A_{\lambda,\sigma}, \quad (9)$$

where $\mathbf{A}$ is the electron-magnetic (external/probe) gauge field, and $\mathbf{a}_\sigma$ is related to the current of spin-σ electrons as follows:

$$j_\sigma^\mu = \frac{1}{2\pi} \epsilon^{\mu\nu\lambda} \partial_\nu a_{\lambda,\sigma}. \quad (10)$$

The above Chern-Simons theory describes a system with $U(1) \times U(1)$ symmetry, where the first $U(1)$ denotes the charge sector, and the second one the spin sector. The above model does not have topological order, meaning that it has a unique ground-state on the torus geometry (periodic system), and contains only trivial (fermionic or bosonic) excitations and does not support any anyon excitation. Now, let us consider a strong onsite attraction between electrons with opposite spins. Such a strong attraction will force the density and current of the two species to become equal at low temperatures. This constraint can be enforced via imposing $a_{\mu,\uparrow}(\mathbf{r}) = a_{\mu,\downarrow}(\mathbf{r}) = b_\mu$. In other words, the interaction gaps out the spin sector via Higgsing the spin gauge field: $a_{s,\mu}(\mathbf{r}) \equiv a_{\mu,\uparrow}(\mathbf{r}) - a_{\mu,\downarrow}(\mathbf{r})$ that can for example be achieved through adding $m_s^2 a_{s,\mu} a_s^\mu$ term to the effective Lagrangian. Implementing the $a_{\mu,\uparrow}(\mathbf{r}) = a_{\mu,\downarrow}(\mathbf{r}) = b_\mu$ constraint in the above effective Lagrangian, we arrive at:

$$\mathcal{L}_{\text{eff}} = \frac{2}{4\pi} \epsilon^{\mu\nu\lambda} b_\mu \partial_\nu b_\lambda - \frac{2e}{2\pi} \epsilon^{\mu\nu\lambda} b_{\mu,\sigma} \partial_\nu A_{\lambda,\sigma}, \quad (11)$$

The above effective $U(1)_2$ Chern-Simons theory describes a liquid of charge $q = 2e$ constituents (doublons) with a on-trivial topological order. Their Hall conductivity is $\sigma_{xy} = \frac{1}{2} \frac{q^2}{h}$, indicating the effective filling fraction (of doublons), ν, equals $1/2$. Moreover, the ground-state of the above theory has a two-fold degeneracy on the torus geometry, and contains only two distinct excitations, namely: trivial (bosnic) and semionic excitations. Hence, all other excitations (including electrons) are confined (i.e., they are highly massive and absent from the effective low energy description).

In our QMC study of this problem, we adopt the staggered π-flux model which is a Chern insulator and has the following tight binding form:

$$H_0 = \sum_{\sigma = \uparrow, \downarrow} \sum_{\mathbf{k}} a_{\mathbf{k}} \left(c_{A,\mathbf{k},\sigma}^\dagger c_{A,\mathbf{k},\sigma} - c_{A,\mathbf{k},\sigma}^\dagger c_{A,\mathbf{k},\sigma} \right)$$
$$+ \sum_{\sigma = \uparrow, \downarrow} \sum_{\mathbf{k}} \left(b_{\mathbf{k}} c_{A,\mathbf{k},\sigma}^\dagger c_{B,\mathbf{k},\sigma} + h.c. \right), \quad (12)$$

where

$$a_{\mathbf{k}} = 2t_2 \left(\cos k_x - \cos k_y \right), \quad (13)$$

and

$$b_{\mathbf{k}} = t_1 e^{-i\pi/4} \left[1 + e^{i(k_y - k_x)} \right] + t_1 e^{i\pi/4} \left[e^{-ik_x} + e^{ik_y} \right] \quad (14)$$

To achieve the maximum flatness of the band-structure, we consider $t_2/t_1 = \sqrt{2}$ (and $t_1 = 1$).

We fill the lower band of the model with spin up and spin down electrons, respectively to achieve the half filling. Next,
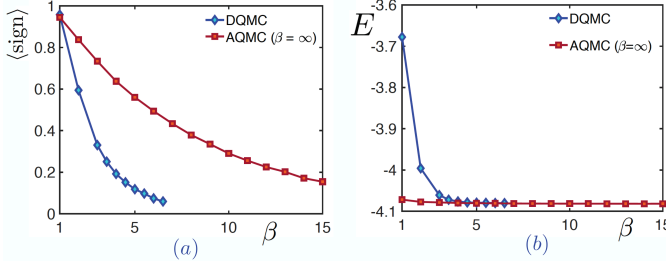
FIG. 13: Comparing the AQMC with DQMC for the correlated Chern insulator model for $U = -6$ on an 8×8 cylinder were we impose a periodic (open) boundary condition along the x (y) direction. The AQMC yields a significantly boosted average sign and a slightly lower energy estimate, which plateaus and must correspond to its true zero temperature value.

we couple electrons via an onsite Hubbard attraction with $U = -6$. Although our model is at half filling and subject to an attractive interaction, it suffers from the sign problem due to the lack of time-reversal symmetry (which is required to guarantee the absence of sign problem in the regular undoped Hubbard model). In Fig. 13, we present the average sign and energy of this model on an 8×8 cylinder.

6. MATLAB IMPLEMENTATION OF AQMC

Below, we provide a simple MATLAB implementation of the AQMC as well as DQMC algorithms. For $\beta_T = 0$, this is an efficient and optimized code whose performance and speed are comparable to C++ and Fortran implementations. For $\beta_T > 0$, to avoid unnecessary programming complications, we break β_T into $N_T = \beta_T / \Delta\tau$ imaginary time steps, instead of a single imaginary time step (i.e., $N_T = 1$, $\Delta\tau = \beta_T$). The other choice, namely $\Delta\tau_T = \beta_T$ will requires several modifications to the code, but instead will reduce the CPU time significantly.

To run this code, copy the main script (marked in blue) and every function (marked in red) in separate MATLAB files. Next, run the main script. In the main script, the user can tune several physical parameters. For example, the onsite Hubbard interaction strength U, the inverse temperature $\beta = T^{-1}$, chemical potential μ, the dimensions of the system along the x (y) direction, N_x (N_y), and the boundary condition along the x (y) axis can be altered easily. A number of other parameters can be tuned as well. For example, the (imaginary) time steps $\Delta\tau$, β_T, $\mathscr{U}_{\max}$, etc can be determined by the user. For the regular DQMC, we just need to set *swtch_adiabatic=0*, and $\beta_T = 0$. For systems with more than about 50 sites, our code automatically employs the delayed update method (which is based on the Sherman-Morrison-Woodbury identity) instead of the standard Sherman-Morrison update rule for the Metropolis algorithm. It is significantly faster for large systems.

This code can be easily parallelized by running the Markov chains in parallel as they are entirely independent. This can be achieved through the *parfor* command in the *func_QMC* function, and the user needs to uncomment the *parpool()* command (or alternatively uses *parpool(nw)*, where *nw* denotes the number of workers (CPUs) available on the computer) at the beginning of the main script.

For the sake of convenience, our code computes the following two-point correlation functions as well: $\langle c^{\dagger}_{\mathbf{i},\sigma} c_{\mathbf{j},\sigma} \rangle$, $\langle s_{z,\mathbf{i}} s_{z,\mathbf{j}} \rangle$, $\langle s_{x,\mathbf{i}} s_{x,\mathbf{j}} \rangle$, and $\langle n_{\mathbf{i},\uparrow} n_{\mathbf{j},\uparrow} \rangle$, provided *swtch_correlation = 1*.

```matlab
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%========================================================================
%              Adiabatic & Determinent Quantum Monte Carlo
%         Simple Matlab code for QMC study of 1D & 2D Hubbard model
%                           Main script
%========================================================================

clc;
clear;
close all;
warning 'off';

% parfor(); % only if parallelization is desired


%========================================================================
%               QMC parameters
%========================================================================

U = 4.0; % Hubbard interaction strength
mu = 0.0; % chemical potential (@ half-filling: mu =0)
n_x = 10; % # of sites along x direction
n_y = 2;  % # of sites along y direction
bnd_x = 1; % boundary condition along x:  0 --> open  1--> periodic
bnd_y = 1; % boundary condition along y:  0 --> open  1--> periodic
beta = 4; % inverse temperature:    beta = 1/T
dtau = 1/10; % imaginary time steps
dk = 10; % stratification period (default = 10)

n_measurement = 5000; % number of measurement sweeps
```

```matlab
n_warmup = round(n_measurement/5); % number of warmup (burn-in) sweeps
n_markov = 6; % number of independent markov chains
% 0 --> only energy is computed
% 1 --> 2-point correlations are also computed
swtch_correlation = 1;

% Adiabatic QMC parameters
swtch_adiabatic = 0; % 0--> regular DQMC 1--> adiabatic QMC
beta_trial = 4; % inverse temperature of trial density matrix
u_max = 4; % u(t) will eventually reach u_max (U) at tau = beta/2
if swtch_adiabatic == 0
    u_max = U;
end
%=====================================================================
%               Hopping amplitudes
%=====================================================================

t_x = 1; % NN hopping along x axis
t_y = t_x; % NN hopping along y axis
t_xy = 0; % NNN hopping
hopping = hopping_matrix(t_x,t_y,t_xy,n_x,n_y,bnd_x,bnd_y);
% adding chemical potential term to enforce the desired electron density
hopping = hopping - mu*diag(ones(1,n_x*n_y));


%=====================================================================
%               QMC algorithm
%=====================================================================

tic;
% func_QMC function contains the QMC algorithm's main steps
[par_func_tmp,correlations_tmp] = func_QMC(swtch_adiabatic,beta_trial,u_max,...
        hopping,beta,n_x,n_y,dtau,dk,n_markov,n_warmup,n_measurement,swtch_correlation);
elapsed_time = toc

%=====================================================================
%               Measurements
%=====================================================================

% working on outputs of func_QMC to generate the desired expectation values
% in expectation_values_computation function
[correlations,measurements] = exp_val(par_func_tmp,correlations_tmp,...
        n_markov,U,mu,n_x,n_y,t_x,t_y,t_xy,bnd_x,bnd_y);


%=====================================================================
%               Plots of two-point correlation functions
%=====================================================================

if swtch_correlation == 1
    x_axis = -round(n_x/2):n_x-round(n_x/2); % distance
    % spin-spin correlation (z component)
    figure(1);
    cor = circshift(correlations{4}{1},-round(n_x/2),2);
    plot(x_axis, [cor,cor(1)],'-o','LineWidth',2);
    xlabel('distance');
    ylabel('G_{s,zz}');
    title('spin-spin correlation (z component)');

    % spin-spin correlation (x component)
    figure(2);
    corr = circshift(correlations{4}{2},-round(n_x/2),2);
    plot(x_axis, [corr,corr(1)],'-d','LineWidth',2);
    xlabel('distance');
    ylabel('G_{s,xx}');
    title('spin-spin correlation (x component)');

    % density-density correlation (for spin up electrons)
    figure(3);
    cor3 = circshift(correlations{4}{3}(1,:),-round(n_x/2),2);
    plot(x_axis,[cor3,cor3(1)] ,'-o','LineWidth',2);
    xlabel('distance');
    ylabel('G_{n}');
    title('pair distribution function');

end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%



%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function hopping = hopping_matrix(t_x,t_y,t_xy,n_x,n_y,bnd_x,bnd_y)

% constructing the matrix of hopping amplitudes for a 2D square lattice
dx = 1;
rx_max = dx; % range of hopping along x
if n_x == 1
    rx_max = 0;
end
dy = 1;
ry_max = dy; % range of hopping along y
if n_y == 1
    ry_max = 0;
end


% hopping amplitudes from sit i to its nearest and next nearest neighbors
T= -[t_xy      t_x    t_xy;
     t_y        0      t_y;
     t_xy      t_x    t_xy];

% hopping matrix
hopping = zeros(n_x*n_y);
for i = 1 : n_x
    for j = 1 : n_y
        % index of site r = (i,j)
        r = (i-1)*n_y + j;
        for rx = -rx_max:rx_max
            ii = i + rx;
            if bnd_x == 1
                ii = 1+mod(ii-1,n_x);
```

```matlab
                end
            for ry = -ry_max:ry_max
                jj = j + ry;
                if bnd_y == 1
                    jj = 1+mod(jj-1,n_y);
                end

                % index of site rr = (ii,jj)
                rr = (ii-1)*n_y + jj;

                if ii>0 && ii <=n_x
                    if jj >0 && jj <=n_y
                        hopping(r,rr) = T(rx+dx+1,ry+dy+1);
                        hopping(rr,r) = hopping(r,rr)';
                    end
                end
            end
        end

    end
    end
end
hopping = (hopping + hopping')/2;

end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [par_func_tmp,correlations_tmp]=func_QMC(swtch_adiabatic,beta_trial,...
    u_max,hopping,beta,n_x,n_y,dtau,dk,n_markov,n_warmup,n_measurement,swtch_corr)

warning 'off';

n_s = n_x*n_y; % total # of sites

n_tau = max(dk,2*round((beta+beta_trial)/(2*dtau))); % # of time steps
dtau = (beta + beta_trial)/n_tau; % time steps
L = round(beta/(2*dtau)); % # of time steps for the evolution period

u_vec = zeros(n_tau,1);
for l = 1:n_tau
    if l*dtau <= beta/2
        u_vec(l) = u_max*(1 - swtch_adiabatic*(l-1)/L);
    elseif l*dtau > (beta/2+beta_trial)
        u_vec(l) = u_max*(1 - swtch_adiabatic*(n_tau-l)/L);
    end
end
lambda = acosh(exp(u_vec*dtau/2));

%========================================================================
% computing expm(-KdT), expm(KdT), expm(-KdT/2), & expm(KdT/2) where K is
% the kinetic term (hopping matrix)
%========================================================================

[U0,D0] = eig(hopping);
D0_d = diag(D0);
B = U0*diag(exp(-dtau*D0_d))*U0';
B_root = U0*diag(exp(-dtau*D0_d/2))*U0';
inv_B = U0*diag(exp(dtau*D0_d))*U0';
inv_B_root = U0*diag(exp(dtau*D0_d/2))*U0';

%========================================================================

num_samples_avg = zeros(n_markov,1);
sign_avg = zeros(n_markov,1);
GF_up_avg = cell(n_markov,1);
GF_dn_avg = cell(n_markov,1);
n_up_dn_avg = cell(n_markov,1);
two_point_corr_avg = cell(n_markov,1);

%========================================================================

parfor count = 1:n_markov

    %====================================================================

    % to be used for Green's function evaluation and space-time wrap
    k_vec = zeros(1,n_tau);
    k_vec(dk:dk:end) = 1;
    k_vec(end) = 1;

    % Initalization:

    num_samples = 0;
    sign_Z = 0;

    two_point_corr = cell(1,3);
    n_up_dn = 0;
    GF_up = 0;
    GF_dn = 0;
    for m = 1 : 3
        two_point_corr{m} = 0 ;
    end
    % to be used for 2-point correlations
    range_corr_x = 1:n_x;
    range_corr_y = 1:n_y;

    %====================================================================

    % random initialization Hubbard-Stratonovich field config.
    s = (2*randi([0,1],n_tau,n_s)-1);
    % h_up = lambda*s, h_dn = -h_up = -lambda*s
    h = diag(lambda)*s;

    % computing corresponding Green's functions for up and down spin electrons
    % computing B_s(l) := exp(-KdT)exp(-h_s(l)) for all l = 1:n_T
    B_up_cell = cluster(B,inv_B,h,n_tau);
    B_dn_cell = cluster(B,inv_B,-h,n_tau);
    % Using QR decomposition to stabilize B_L ... B_1
    % to compute G = inv(Id + B_L ... B_1)  for B = B_up or B_dn
    [G_up,G_dn,log_det_G,sign_det_G] = initial_eval(B_up_cell,B_dn_cell,k_vec,n_tau);
```

```matlab
% the sign of various HS field configs is needed for computing
% expectation values
sign_Z_accumulated = sign_det_G;

%========================================================================

% space-time sweeps:
for i_swp = 1:(n_warmup+n_measurement)
    for l = 1: n_tau
        % Green's function update upon Hubbard-Stratonovich fields flip
        % for large systems delayed update (Sherman-Morrison-Woodbury) is faster
        % than Sherman-Morrison update method
        if n_s > 50
            delayed_period = 32;
            [G_up,G_dn,h,sign_Z_accumulated_tmp] = delayed_update(G_up,G_dn,h,n_s,delayed_period);
        else
            [G_up,G_dn,h,sign_Z_accumulated_tmp] = Sher_Mor(G_up,G_dn,h,n_s);
        end
        % sign_current = sign_prev*sign_tmp
        sign_Z_accumulated = sign_Z_accumulated*sign_Z_accumulated_tmp;
        % space time wrap up by one time-step
        h = circshift(h,-1,1);
        k_vec = circshift(k_vec,-1,2);

        if k_vec(1) == 1
            % reevaluating the Green's function from scratch
            B_up_cell = cluster(B,inv_B,h,n_tau);
            B_dn_cell = cluster(B,inv_B,-h,n_tau);
            [G_up,G_dn,~,~] = initial_eval(B_up_cell,B_dn_cell,k_vec,n_tau);

        else
            % spacetime wrapping using already existing Green's functions
            h_tmp = h(n_tau,:);

            % spin up
            B_up_prev = B*diag(exp(h_tmp));
            inv_B_up_prev = diag(exp(-h_tmp))*inv_B;
            G_up = B_up_prev*G_up*inv_B_up_prev;

            % spin dn
            B_dn_prev = B*diag(exp(-h_tmp));
            inv_B_dn_prev = diag(exp(h_tmp))*inv_B;
            G_dn = B_dn_prev*G_dn*inv_B_dn_prev;
        end

    end

    %=============================================
    % Measurements for the current HS field config.
    %=============================================

    if  i_swp > n_warmup
        num_samples = num_samples + 1;

        sign_Z = sign_Z + sign_Z_accumulated;

        % inv_B_root*G_up*B_root & inv_B_root*G_dn*B_root
        % instead of G_up and G_dn are used to employ the
        % 2nd order Trotter-Suzuki decomposition
        GF_up_tmp = eye(n_s) - inv_B_root*G_up*B_root;
        GF_dn_tmp = eye(n_s) - inv_B_root*G_dn*B_root;

        % density profile for the current HS fields
        n_up_tmp = diag(GF_up_tmp);
        n_dn_tmp = diag(GF_dn_tmp);

        % spin and total density for the current HS fields
        sz_tmp  = (n_up_tmp-n_dn_tmp)/2;
        rho_tmp = (n_up_tmp+n_dn_tmp)/2;

        % to compute average GF and <n_up n_dn>
        GF_up = GF_up + GF_up_tmp*sign_Z_accumulated;
        GF_dn = GF_dn + GF_dn_tmp*sign_Z_accumulated;
        n_up_dn = n_up_dn + n_up_tmp.*n_dn_tmp*sign_Z_accumulated;

        % two-point correlation functions: spin-spin
        % and density-density correlation functions along x axis
        % namely: <O(x0,y0)O(x0+x,y0)>
        if swtch_corr == 1
            corr_tmp = cell(1,3);

            corr_0 = cell(1,3);
            for m = 1 : 3
                corr_0{m} = 0;
            end
            count_tmp = 0;
            for ix_1 = range_corr_x
                for iy_1 = range_corr_y
                    r_1 = (ix_1-1)*n_y + iy_1; % first coordinate of the correlation function
                    r_2 = n_y*[(ix_1:n_x) (1:(ix_1-1))] - n_y + iy_1; % 2nd coordinate of the correlation function

                    %  <sz(r_1)sz(rr_2)> computation
                    corr_tmp{1} = (sz_tmp(r_1)*sz_tmp(r_2)).'- (GF_up_tmp(r_1,r_2).*GF_up_tmp(r_2,r_1).')/4 ...
                            -(GF_dn_tmp(r_1,r_2).*(GF_dn_tmp(r_2,r_1)).')/4;

                    corr_tmp{1}(1) = -n_up_tmp(r_1)*n_dn_tmp(r_1)/2 + (rho_tmp(r_1))/2;

                    %  <sx(r_1)sx(r_2)> computation
                    corr_tmp{2} = -GF_up_tmp(r_1,r_2).*(GF_dn_tmp(r_2,r_1).')/4 - GF_dn_tmp(r_1,r_2).*(GF_up_tmp(r_2,r_1).')/4;

                    corr_tmp{2}(1) = -n_up_tmp(r_1)*n_dn_tmp(r_1)/2 +(rho_tmp(r_1))/2;

                    %  <n_{up}(r_1)n_{up}(r_2)> computation
                    corr_tmp{3} =  n_up_tmp(r_1)*n_up_tmp(r_2).'- (GF_up_tmp(r_1,r_2).*GF_up_tmp(r_2,r_1).');
                    corr_tmp{3}(1) = 0;


                    for m = 1 : 3
                        corr_0{m} = corr_0{m} + corr_tmp{m};
                    end
                    count_tmp = count_tmp + 1;
```

```
                                        end
                                    end
                                    for m = 1 : 3
                                        corr_0{m} = corr_0{m}/count_tmp;
                                    end

                                    for m = 1 : 3
                                        two_point_corr{m} = two_point_corr{m} + corr_0{m}*sign_Z_accumulated;
                                    end

                            end
                        end
                    end

                % <O> =  <Osign(z)>/<sign(z)>
                GF_up_avg{count} = GF_up/sign_Z;
                GF_dn_avg{count} = GF_dn/sign_Z;
                n_up_dn_avg{count} = n_up_dn/sign_Z;
                for m = 1:3
                    two_point_corr_avg{count}{m} = two_point_corr{m}/sign_Z;
                end
                num_samples_avg(count) = num_samples;
                % average sign of fermion determinants, Z, where
                % Z = 1/(Det(G_up)Det(G_dn)) for a given Hubbard-Stratonvich field
                sign_avg(count) = sign_Z/num_samples;
        end

        % expectation values for each Markov chain is stored in a cell structure
        % for final statistical average in a separate function
        correlations_tmp = cell(1,4);
        correlations_tmp{1} = GF_up_avg;
        correlations_tmp{2} = GF_dn_avg;
        correlations_tmp{3} = n_up_dn_avg;
        correlations_tmp{4} = two_point_corr_avg;

        % saving average spin and of number of samples related to partition
        % function in another cell structure.
        par_func_tmp = cell(1,2);
        par_func_tmp{1} = num_samples_avg;
        par_func_tmp{2} = sign_avg;

    end
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    function [B,inv_B] = cluster(B_k,inv_B_k,h,n_l)

    % Here we compute B_s(l) = exp(-Kdt)exp(-h_s(l)), h_up = -h_dn = h
    % We also compute inv(B_s(l)) = exp(h_s(l))exp(Kdt)
    h = sparse(h);
    B = cell(n_l,1);
    inv_B = cell(n_l,1);
    for l = 1:n_l
        h_tmp = h(l,:);
        B{l} = B_k*diag(exp(h_tmp));
        inv_B{l} = diag(exp(-h_tmp))*inv_B_k;
    end

    end
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    function [G_p,G_m,log_det_G,sign_det_G] = initial_eval(B_up_hat,B_dn_hat,k_vec,n_l)

    % Here we use startification & QR decomposition
    % to compute the GF from scratch

    % spin up:
    [G_p,log_det_G_p,sign_det_G_p] = strat(B_up_hat,k_vec,n_l);

    % spin down:
    [G_m,log_det_G_m,sign_det_G_m] = strat(B_dn_hat,k_vec,n_l);

    % log of asb val of det(G_up*G_dn)
    log_det_G = log_det_G_p + log_det_G_m;
    % sign of abs val of det(G_dn*G_dn)
    sign_det_G = sign_det_G_p*sign_det_G_m;

    end
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    function [G,log_det_G,sign_det_G] = strat(B,k_vec,n_l)

    % computing Green's function via G = inv(Id + B_L...B_1) using QR
    % decomposition for stabilization
    % We also compute the sign and (log of ) abs value of GF's determinant

    Q = 1;
    D = 1;
    T = 1;
    i = 1;
    while i <= n_l

        % stratification (i.e. multiplying dk B matrices directly
        % (no QR is needed as long as dk is small e.g. dk = 8 or 10))
        B0 = B{i};
        i = i +1;
        while i <= n_l && k_vec(i) < 1
            B0 = B{i}*B0;
            i = i + 1;
        end
        c = (B0*Q)*D;

        % QR decomposition to multiply c to the accumulated
        % product of c (and as a result B matrices) up to the current point
        [Q,R] = qr(c);
        D0 = sparse(diag(R));
```

```
    inv_D = diag(1./D0);
    D = diag(D0);
    T = (inv_D*R)*T;

end

% Using D, T, and Q we finally compute
% GF with special care (to avoid numerical instability)
D_diag = diag(D);
D_b = max(1,abs(D_diag)).*sign(D_diag);
D_s = min(1,abs(D_diag));

inv_D_b = diag(sparse(1./D_b));
D_s = diag(sparse(D_s));

a1 = inv_D_b*(Q.') + D_s*T;
a2 = inv_D_b;
a3 = (Q.');

G = (a1\a2) * a3;


% computing log(det(abs(G))) as well as sign(det(G)) accurately
% and with special care (again to avoid numerical instabilities)
[L1,U1] = lu(a1);

log_det_L1 = 0;
sign_det_L1 = sign(det(L1));
log_det_U1 = sum(log(abs(diag(U1))));
sign_det_U1 = prod(sign(diag(U1)));
log_det_a1 = log_det_L1 + log_det_U1;
sign_det_a1 = sign_det_L1*sign_det_U1;

log_det_a2 = -sum(log(abs(D_b)));
sign_det_a2 = prod(sign(D_b));

log_det_a3 = 0;
sign_det_a3 = sign(det(a3));

sign_det_G = sign_det_a1*sign_det_a2*sign_det_a3; % sign of det(G)
log_det_G = -log_det_a1+log_det_a2+log_det_a3; % log of abs(det(G))

end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [G_up,G_dn,h,sgn_accumulated_tmp] = Sher_Mor(G_up,G_dn,h,n_s)

id = eye(n_s);
sgn_accumulated_tmp = 1;

for i = 1:n_s
    if h(1,i) ~= 0
        % ratio of 1/det(G_up)
        alpha_up = exp(-2*h(1,i))-1;
        r_up = 1+alpha_up*(1-G_up(i,i));

        % ratio of 1/det(G_dn)
        alpha_dn = exp(+2*h(1,i))-1;
        r_dn = 1+alpha_dn*(1-G_dn(i,i));

        % total ratio
        %=====================================
        r = r_up*r_dn;
        %=====================================

        % Hubbard-Stratonovic field flip acceptance based on Metropolis-Hastings algorithm
        % & as a consequence Green's function update based on the
        % Sherman-Morrison update formula
        if rand(1) <= abs(r)

            % spin up GF update:
            a_up = (id - G_up);
            b_up = G_up;
            G_up = G_up - (alpha_up/r_up)*a_up(:,i)*b_up(i,:);

            % spin dn GF update:
            a_dn = (id - G_dn);
            b_dn = G_dn;
            G_dn = G_dn - (alpha_dn/r_dn)*a_dn(:,i)*b_dn(i,:);

            % update h of accumulated sign of det
            h(1,i) = -h(1,i);
            sgn_accumulated_tmp = sgn_accumulated_tmp * sign(r);
        end
    end
end

end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [G_up,G_dn,h,sgn_accumulated_tmp] = delayed_update(G_up,G_dn,h,n_s,delayed_period)

% right update
id = eye(n_s);
sgn_accumulated_tmp = 1;

% right update
a_up = zeros(n_s,delayed_period);
b_up = zeros(delayed_period,n_s);
d_up = diag(G_up);
a_dn = a_up;
b_dn = b_up;
d_dn = diag(G_dn);

k = 1;
for i = 1:n_s
    if h(1,i) ~= 0
        % ratio of 1/det(G_up)
```

```
        alpha_up = exp(-2*h(1,i))-1;
        r_up = 1+alpha_up*(1-d_up(i));

        % ratio of 1/det(G_dn)
        alpha_dn = exp(2*h(1,i))-1;
        r_dn = 1+alpha_dn*(1-d_dn(i));

        % total ratio
        %============================================================
        r = r_up*r_dn;
        %============================================================

        % Hubbard-Stratonic field flip acceptance based on Metropolis-Hasting algorithm
        % and as a consequence Green's function update based on the
        % Sherman-Morrison-Woodbury update formula
        if rand(1) <= abs(r)
            v1 = 1:(k-1);

            % spin up GF update:
            a_up(:,k) = G_up(:,i)-id(:,i) + a_up(:,v1)*b_up(v1,i);
            a_up(:,k) = (alpha_up/r_up)*a_up(:,k);
            b_up(k,:) = G_up(i,:) + a_up(i,v1)*b_up(v1,:);
            d_up = d_up + a_up(:,k).*b_up(k,:).';

            % spin dn GF update:
            a_dn(:,k) = G_dn(:,i) -id(:,i)+ a_dn(:,v1)*b_dn(v1,i);
            a_dn(:,k) = (alpha_dn/r_dn)*a_dn(:,k);
            b_dn(k,:) = G_dn(i,:) + a_dn(i,v1)*b_dn(v1,:);
            d_dn = d_dn + a_dn(:,k).*b_dn(k,:).';

            k = k+1;
            h(1,i) = -h(1,i);

            sgn_accumulated_tmp = sgn_accumulated_tmp * sign(r);

        end
        if k == delayed_period+1
            G_up = G_up + a_up*b_up;
            d_up = diag(G_up);

            k = 1;

            G_dn = G_dn + a_dn*b_dn;
            d_dn = diag(G_dn);

        end
    end
end

v1 = 1:(k-1);
G_up = G_up + a_up(:,v1)*b_up(v1,:);
G_dn = G_dn + a_dn(:,v1)*b_dn(v1,:);

end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [correlations,measurements] = exp_val(par_func_tmp,...
        correlations_tmp,n_markov,U,mu,n_x,n_y,t_x,t_y,t_xy,bnd_x,bnd_y)

n_s = n_x*n_y; % number of sites
hopping = hopping_matrix(t_x,t_y,t_xy,n_x,n_y,bnd_x,bnd_y);% hopping matrix

% combining results of all Markov chains
GF_up_avg = correlations_tmp{1};
GF_dn_avg = correlations_tmp{2};
n_up_dn_avg = correlations_tmp{3};
two_point_corr_avg = correlations_tmp{4};

sign_avg = par_func_tmp{2};

energy_avg = zeros(n_markov,1);
for count = 1:n_markov
    % average ground-state E
    energy_avg(count,1) = (U*sum(n_up_dn_avg{count})+...
        trace(hopping*(GF_up_avg{count}+GF_dn_avg{count})))/n_s;
end

energy_mean = 0;
sign_mean = 0;
GF_up_mean = 0;
GF_dn_mean = 0;
n_up_dn_mean = 0;
two_point_corr_mean = cell(1,3);
for l = 1:3
    two_point_corr_mean{l} = 0;
end

for count = 1:n_markov
    energy_mean = energy_mean + energy_avg(count,1);
    sign_mean = sign_mean + abs(sign_avg(count,1));
    GF_up_mean = GF_up_mean + GF_up_avg{count};
    GF_dn_mean = GF_dn_mean + GF_dn_avg{count};
    n_up_dn_mean = n_up_dn_mean + n_up_dn_avg{count};
    for l = 1:3
        two_point_corr_mean{l}=two_point_corr_mean{l}+...
            two_point_corr_avg{count}{l};
    end
end

energy_mean = energy_mean/n_markov;
sign_mean = sign_mean/n_markov;
GF_up_mean = GF_up_mean/n_markov;
GF_dn_mean = GF_dn_mean/n_markov;

n_up_mean = diag(GF_up_mean);
n_dn_mean = diag(GF_dn_mean);
n_up_dn_mean = n_up_dn_mean/n_markov;
for l = 1:3
    two_point_corr_mean{l} = two_point_corr_mean{l}/n_markov;
end
```

```matlab
% saving results in a cell structure
correlations = cell(1,10);
correlations{1}{1} = sign_avg;
correlations{2}{1} = GF_up_mean;
correlations{2}{2} = GF_dn_mean;
correlations{3} = n_up_dn_mean;
correlations{4} = two_point_corr_mean;
correlations{5} = energy_avg;

% average density
n_mean = mean(n_up_mean)+mean(n_dn_mean);

% average kinetic energy
kin_energy = energy_mean -U*mean(n_up_dn_mean);

% average Hubbard interaction energy
int_energy = U*mean(n_up_dn_mean);

% normalized statistical error bar of DQMC calculations
err_bar = 10^2*std(energy_avg)/abs(mean(energy_avg));

% most important measurements

fprintf('measurements=[density,energy,kin_energy,int_energy,err_bar,<sign>]');

measurements=[n_mean,energy_mean,kin_energy,int_energy,err_bar,sign_mean]

end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```