Supplemental Material for "Efficient simulations of Hartree–Fock equations by an accelerated gradient descent method"

by Y. Ohno, A. Del Maestro, and T.I. Lakoba

I. DERIVATION OF THE SECOND TERM IN EQ. (6b)

In Sec. 2 of [S1] it was shown that when minimizing a functional $\mathcal{H}$ such that $\delta H/\delta\phi_j^* = L_{00}\phi_j$ subject to constraints given by equations $Q^{(k)} = 0$, $k = 1, \ldots, K$, the second term in Eq. (6b)[S2] is given by:

$$\Phi \, \langle P^{-1}\Phi \,|\, \Phi\rangle^{-1} \, \langle P^{-1}\Phi \,|\, \overrightarrow{L_{00}\phi}^{\,T}\rangle. \tag{S1}$$

Here the (j,k)-th component of the $N \times K$ matrix Φ is:

$$\Phi_{j,k} = \delta Q^{(k)}/\delta\phi_j, \tag{S2}$$

$$\overrightarrow{L_{00}\phi} = [L_{00}\phi_1, \ldots, L_{00}\phi_N]\,, \tag{S3}$$

the rest of the notations — here and below — are defined in Sec. II A of the main text, and we have omitted iteration's index n.

For the constraints Q given by the $K = N^2$ Eqs. (2), definition (S2) yields:

$$\Phi = \vec{\phi} \otimes I, \tag{S4}$$

where $\otimes$ stands for the Kronecker (tensor) product and I is the identity matrix of appropriate dimension ($N \times N$ in this case). Then $P^{-1}\Phi = P^{-1}\vec{\phi} \otimes I$. Next, using the identities (for matrices A, B, C, D of appropriate dimensions):

$$(A \otimes B)\,(C \otimes D) = (AC) \otimes (BD), \qquad (A \otimes B)^{-1} = A^{-1} \otimes B^{-1}, \tag{S5}$$

one has

$$\langle P^{-1}\Phi \,|\, \Phi\rangle^{-1} = \langle P^{-1}\vec{\phi}\,|\,\vec{\phi}\rangle^{-1} \otimes I \equiv \begin{pmatrix} a_{11}\cdot I & \cdots & a_{1N}\cdot I \\ \vdots & \vdots & \vdots \\ a_{N1}\cdot I & \cdots & a_{NN}\cdot I \end{pmatrix}, \tag{S6}$$

with a_{ij} denoting entries of the inverse matrix in the middle of the above line. To proceed, one needs to

write out an explicit form of the last factor in (S1):

$$\langle P^{-1}\Phi \,|\, \overrightarrow{L_{00}\phi}^{\,T}\rangle = \begin{pmatrix} \langle P^{-1}\phi_1 \,|\, L_{00}\phi_1\rangle \\ \vdots \\ \langle P^{-1}\phi_1 \,|\, L_{00}\phi_N\rangle \\ \vdots \\ \langle P^{-1}\phi_N \,|\, L_{00}\phi_1\rangle \\ \vdots \\ \langle P^{-1}\phi_N \,|\, L_{00}\phi_N\rangle \end{pmatrix}. \tag{S7}$$

Multiplying the second and third factors in (S1) and using (S6), (S7) yields:

$$\langle P^{-1}\Phi \,|\, \Phi\rangle^{-1} \langle P^{-1}\Phi \,|\, \overrightarrow{L_{00}\phi}^{\,T}\rangle = \begin{pmatrix} \sum_{k=1}^{N} a_{1k}\langle P^{-1}\phi_k \,|\, L_{00}\phi_1\rangle \\ \vdots \\ \sum_{k=1}^{N} a_{1k}\langle P^{-1}\phi_k \,|\, L_{00}\phi_N\rangle \\ \vdots \\ \sum_{k=1}^{N} a_{Nk}\langle P^{-1}\phi_k \,|\, L_{00}\phi_1\rangle \\ \vdots \\ \sum_{k=1}^{N} a_{Nk}\langle P^{-1}\phi_k \,|\, L_{00}\phi_N\rangle \end{pmatrix}. \tag{S8}$$

Finally, using (S4), one can see by inspection that j-th row of the entire quantity in (S1) is given by:

$$\sum_{i=1}^{N} \phi_i \sum_{k=1}^{N} a_{ik}\langle P^{-1}\phi_k \,|\, L_{00}\phi_j\rangle, \tag{S9}$$

which is indeed the second term on the r.h.s. of (6b).

II. EQUATIONS $L_0\phi_j = 0$ AND $L^{(0)}\phi_j = 0$ ARE EQUIVALENT

The operator in the second equation in the section title is defined in (6b). Now, the former equation, i.e., (3), can be written as

$$L_{00}\phi_j = \vec{\phi}\,\langle\vec{\phi}\,|\, L_{00}\phi_j\rangle\,. \tag{S10}$$

Similarly, the latter equation is written as:

$$L_{00}\phi_j = \vec{\phi}\,\langle P^{-1}\vec{\phi}\,|\vec{\phi}\,\rangle^{-1}\,\langle P^{-1}\vec{\phi}\,|\, L_{00}\phi_j\rangle\,. \tag{S11}$$

In addition, constraints (2) can be written as

$$\langle\vec{\phi}\,|\,\vec{\phi}\,\rangle = I. \tag{S12}$$

Let us denote $\langle P^{-1}\vec{\phi}\,|\vec{\phi}\,\rangle \equiv M$ (an $N \times N$ matrix).

The proof consists of showing that (S10) implies (S11) and vice versa. To show the former step, substitute (S10) into the r.h.s. of (S11); this yields:

$$\vec{\phi}\,M^{-1}\left\langle\, P^{-1}\vec{\phi}\,|\,\vec{\phi}\,\langle\vec{\phi}\,|\, L_{00}\phi_j\rangle\,\right\rangle = \vec{\phi}\,M^{-1}\,M\,\langle\vec{\phi}\,|\, L_{00}\phi_j\rangle, \tag{S13}$$

which, per (S10), equals the l.h.s. of (S11).

To show that (S11) along with (S12) implies (S10), substitute the r.h.s. of (S11) into the r.h.s. of (S10) while multiplying the left (right) part of the inner product by P^{-1} (P) and using the fact that P is Hermitian. Then the r.h.s. of (S10) becomes:

$$\vec{\phi}\left\langle P^{-1}\vec{\phi}\,|\,P\,\vec{\phi}\,M^{-1}\langle P^{-1}\vec{\phi}\,|\,L_{00}\phi_j\rangle\,\right\rangle = \vec{\phi}\,\langle P^{-1}\vec{\phi}\,|\,P\,\vec{\phi}\,\rangle\,M^{-1}\langle P^{-1}\vec{\phi}\,|\,L_{00}\phi_j\rangle = \vec{\phi}\,M^{-1}\langle P^{-1}\vec{\phi}\,|\,L_{00}\phi_j\rangle, \quad \text{(S14)}$$

where we have used (S12). Per (S11), this equals the l.h.s. of (S10).

III. DIFFERENCES OF THE AITEM FROM THE SELF-CONSISTENT FIELD METHOD (SCF) USED IN NUCLEAR AND ELECTRONIC STRUCTURE CALCULATIONS

For the sake of simplifying this discussion, we will assume no preconditioning in Eqs. (6); i.e., $P = I$ in Eq. (7). Then the expression for $(L^{(0)}\phi_j)_n$ reduces to $(L_0\phi_j)_n$ in Eq. (3).

At iteration n of the AITEM, one uses $(L_0\phi_j)_n$ (and, if mode elimination is used, also the Γ-term in (6a)) to advance the solution $\vec{\phi}$ to iteration $(n+1)$, with the goal being solving the HF equations (3). Iterations are terminated when $\|\overrightarrow{(L_0\phi_j)}_n\|$ decreases below a specified threshold.

In the SCF, the solution of the HF equations (or the Density Functional Theory equations) proceeds differently (see, e.g., [S3] for the coordinate-based nuclear structure calculations or [S4] for electronic structure calculations using an atomic orbitals basis). First, one recognizes that a unitary transformation $\vec{\phi}\,' = S\vec{\phi}$ can diagonalize the matrix of Lagrange multipliers $\langle\phi_i|L_{00}\phi_j\rangle$ while preserving the orthonormality relations (2) and the form of $\overrightarrow{L_{00}\phi}$ [S4]. Then Eqs. (3) at iteration n are transformed into a nonlinear eigenvalue problem

$$\overrightarrow{(L_{00}\phi')}_n = \epsilon_n(\vec{\phi}\,')_n, \quad \text{(S15)}$$

which is solved self-consistently as follows. One rewrites it as

$$(L_{00})_n\,(\vec{\phi}\,')_{n+1} = \epsilon_{n+1}(\vec{\phi}\,')_{n+1}, \quad \text{(S16)}$$

where $(L_{00})_n$ is evaluated on $(\vec{\phi}\,')_n$. Then one finds (also by some iterative algorithm) N_1 eigenstates of the *linear* eigenvalue problem (S16), where one usually sets $N_1 > N$. From them, one selects N eigenstates $(\vec{\phi}\,')_{n+1}$ with lowest ϵ, which are orthonormal as eigenstates of a Hermitian operator $(L_{00})_n$, and uses them at the next iteration. The iterations are terminated when $\|(\vec{\phi}\,')_{n+1} - (\vec{\phi}\,')_n\|$ decreases below a prescribed threshold. The fact that, in theory, both the ITEM (without acceleration) and SCF converge to the same ground state of the HF equations was shown in [S5]. In [S6] it was shown that, in practice, the ITEM may converge to a lower-energy state than the SCF.

Note that the ability to initially compute $N_1 > N$ eigenstates of operator $(L_{00})_n$ hinges on the fact that its dimension, which equals the number of grid points, is (much) greater than the number N of simulated

particles. (For the methods based on atomic orbitals bases, one simply needs to choose the number of the orbitals to be greater than the number of particles.) Using $N_1 > N$ is needed to avoid or suppress some undesirable artifacts, similar to the error oscillations reported in Sec. III C, which are due to closely lying energy levels ϵ in (S16). In contrast, the number of particles in the AITEM described in Sec. II B cannot be altered without simultaneously altering the physical system, governed by Eqs. (3).

Expanding on the latter point, we will now explain why using the SCF *with* $N_1 > N$, which is fully appropriate in the contexts of the nuclear or electronic structure calculations, would not be appropriate in the context of atoms scattered over a surface. The electrons in an atom satisfy the so-called Aufbau Principle, which says that in the ground state of an atom, the electrons fill atomic orbitals of the lowest available energy levels before occupying higher-energy ones; that is, the orbitals are filled in the increasing order of their energies. Thus, the energy levels and the eigenstates of an N-electron atom/ion are a slightly modified *subset* of the energy levels/eigenstates of the $N_1 > N$-electron atom/ion. In contrast, atoms scattered as a monolayer over a surface have their energies determined by the *entire pattern* that they form. In particular, the sets of energies $\{\epsilon_j\}_{j=1}^{N}$, where $\epsilon_j \equiv \langle\phi_j|L_{00}|\phi_j\rangle$, of N scattered atoms will not be a (possibly slightly modified) subset of the energies of $N_1 > N$ scattered atoms.

We will illustrate the above statement with the 1D Hartree-only problem, given by Eqs. (14), (15) of Sec. III. (The behavior of the AITEM is qualitatively similar for that model and for the full HF model.) In Fig. S1 we show solutions of systems with $N = 15, 16, 17$ atoms in a computational window containing 23 potential wells. The latter number is chosen to be prime so that $N/23$ be an irreducible fraction, while N is sufficiently large so that incrementing it by 1 may intuitively appear as a small change to the system, but sufficiently small so that the repulsion between adjacent atoms is not too strong.

One can see that the solution for the $N = 15$ system are *different* from atoms 1–15 of the $N = 16$ or $N = 17$ systems: namely, the latter occupy different positions. The positions of the atoms, along with their shape, are what determine the solution of the Hartree equations. Thus, the solution of the $N = 15$ system is *not a subset* of the $N = 16$ or $N = 17$ systems.

The same fact can also be seen from the energies ϵ_j of the three systems, which are listed in Table S1. Namely, while there are similarities between some energies of the $N = 15$ system and those of the $N = 16$ systems, the former energies are far from being a slightly modified subset of the latter. Moreover, all those similarities disappear for $N = 17$, where *all* of ϵ_j are strictly greater than those for $N = 15$.

Let us now show that the (A)ITEM and SCF have the same computational complexity in the order-of-magnitude sense. From the brief descriptions of these methods at the beginning of this section, one sees that their main computational difference is as follows: In the AITEM, one needs to compute $N(N+1)/2$ inner products $\langle\phi_i|L_{00}\phi_j\rangle$, whereas in the SCF one needs to find $N1 \gtrsim N$ orthonormal eigenstates of operator

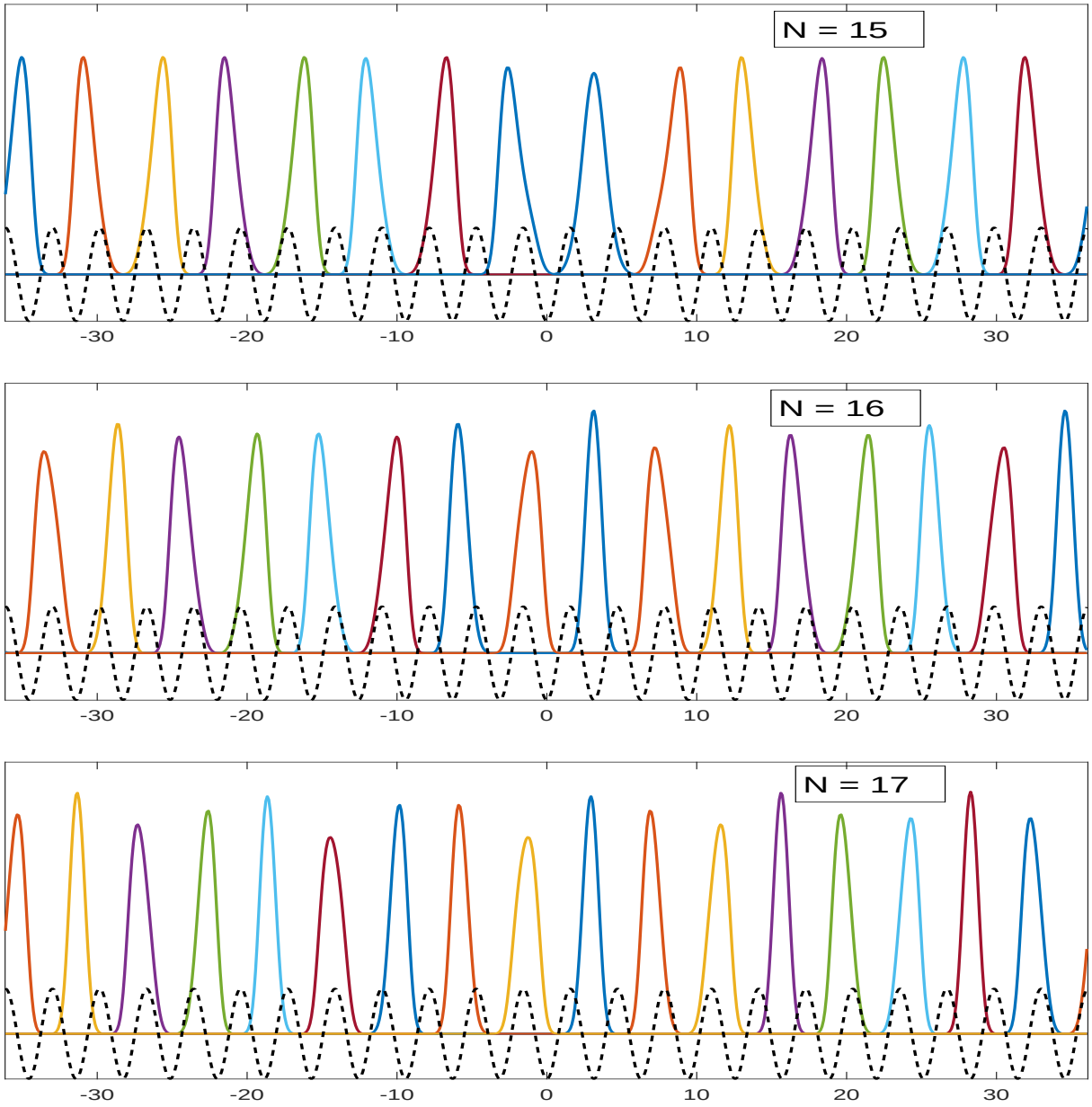

FIG. S1. Colored lines: Solutions ϕ_j of the Hartree equations (14), (15) with 23 wells of $V_{\rm ext}$ for $N = 15, 16, 17$ helium atoms, as indicated. Dashed line: $V_{\rm ext}/10$, shown for reference.

N	ϵ_j																
15	0.20	0.20	0.20	0.20	0.19	0.22	0.08	0.38	-0.68	0.38	0.08	0.22	0.19	0.20	0.20	—	—
16	1.28	0.04	0.46	0.20	0.20	0.46	0.04	1.28	0.05	1.18	0.08	0.38	0.38	0.08	1.18	0.05	—
17	1.15	0.42	1.47	0.83	0.35	1.65	0.43	0.43	1.65	0.35	0.83	1.47	0.42	1.15	1.32	0.44	1.32

TABLE S1. The energies ϵ_j of atoms ϕ_j in the panels of Fig. S1, listed from the leftmost to the rightmost atom.

L_{00}. Each of these steps has

$$\begin{pmatrix}\text{Computational}\\ \text{complexity}\end{pmatrix} = O(N^2) \times \begin{pmatrix}\text{Complexity of}\\ \text{computing } L_{00}\phi_j\end{pmatrix}. \tag{S17}$$

Since terms in $L_{00}\phi_j$ contain operations diagonal either in the physical space or the momentum space and

convolutions, the second factor on the r.h.s. of (S17) is $O(N) \times M_{\rm grid} \log M_{\rm grid}$ when the Fock (i.e., the exchange interaction) term is not included in (4) and $O(N^2) \times M_{\rm grid} \log M_{\rm grid}$ when it is included (see, however, Remark Six at the end of Sec. II B); here $M_{\rm grid}$ is the number of points on the computational grid. While the order of magnitude of the computational complexity of both methods is the same, it was shown in [S6] that in many cases, the ITEM was some factor of 2–3 slower than the SCF. However, the ITEM in [S6] was not accelerated by either preconditioning or mode elimination, as it is done in our work.

IV. EFFECT OF PRECONDITIONER ON CONVERGENCE RATE OF THE AITEM

Here we complement the discussion found in Sec. III A and show that, depending on the spatial discretization step, preconditioner P of Eq. (7) with a suboptimal c may even degrade convergence rate when compared to the case where no preconditioner is used (i.e., $P = I$, the identity matrix). On the contrary, the preconditioner with an optimal c always improves the convergence rate of the AITEM for all practically relevant values of the discretization step.

As follows from Eq. (13), the convergence rate is proportional to $(\lambda_{P^{-1}L})_{\rm min}/(\lambda_{P^{-1}L})_{\rm max}$ (when this quantity is small). When no preconditioner is used, $(\lambda_L)_{\rm max} = O(k_{\rm max}^2 + E_{\rm clip}) = O\big(10/(\Delta x)^2 + E_{\rm clip}\big)$; see Sec. III A. Here '10' approximates π^2 in $k_{\rm max}^2 = \pi^2/(\Delta x)^2$. Assuming that $(\lambda_L)_{\rm min} = O(1)$, as confirmed by the calculations found after Eq. (17), this predicts the convergence rate to decrease as $O\big((\Delta x)^2\big)$ for sufficiently small Δx. Panel (a) of Fig. S2 confirms this prediction in the order-of-magnitude sense. There, $\Delta x = \pi/M_{x1}$, where now π is the period of $V_{\rm ext}$ used in Sec. III and M_{x1} is the number of discretization points per this period. For the data in Fig. S2(a) M_{x1} was chosen as that in the 2D simulations reported in Sec. V ($M_{x1} = 16$), as in the 1D results reported in all Figures of Sec. III A except Fig. 2 ($M_{x1} = 32$), and as in Fig. 2 ($M_{x1} = 64$).

On the other hand, when P is given by (7), the ratio $(\lambda_{P^{-1}L})_{\rm min}/(\lambda_{P^{-1}L})_{\rm max}$ cannot be estimated as simply as above: see, again, the calculations found after Eq. (17). One can only conclude that the strong dependence of that ratio (and hence of the convergence rate) on Δx is suppressed by P, because

$$(\lambda_{P^{-1}L})_{\rm max} = \frac{O(k_{\rm max}^2 + E_{\rm clip})}{O(k_{\rm max}^2 + c)} \sim \frac{1 + O\big(E_{\rm clip}(\Delta x)^2/10\big)}{1 + O\big(c\,(\Delta x)^2/10\big)}.$$

In fact, our simulations of the setup of Fig. S2(a) but with P given by (7) with $c = 2$ or $c = E_{\rm clip} = 100$ revealed that the number of iterations to convergence remains the same to three significant figures when M_{x1} is varied as in that Figure. In Fig. S2(b) we show these results only for $M_{x1} = 32$, as those for $M_{x1} = 16$ and $M_{x1} = 64$ look indistinguishable from them.

Comparison of Figs. S2(a) and (b) shows that for $M_{x1} = 32$ and 16, the ITEM without the preconditioner is faster than the AITEM using P with the suboptimal $c = 2$. On the other hand, the AITEM with the

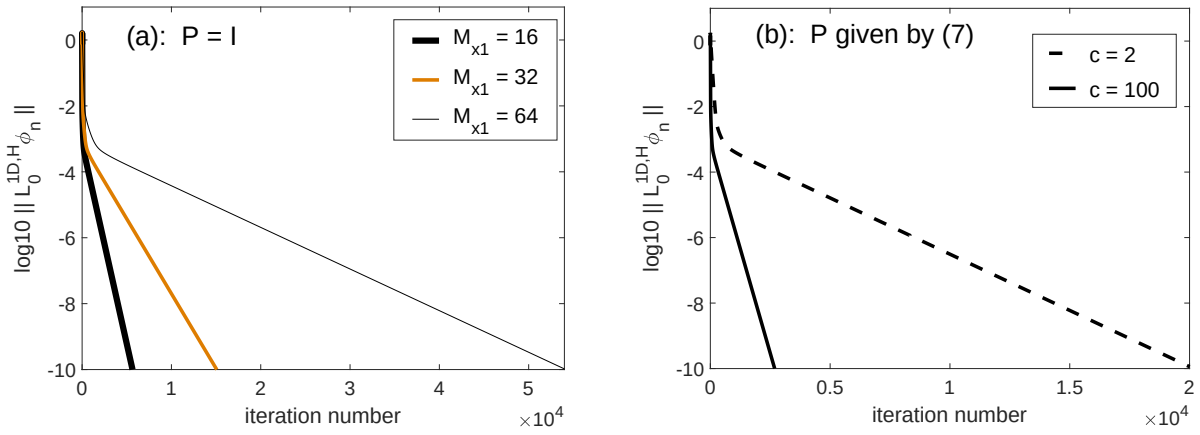

FIG. S2. Error evolution of the AITEM (6) with $P = I$ (a) and with P given by Eq. 7 (b) for the setup of Fig. 4 with no slow-mode elimination. Parameter Δt of the (A)ITEM was optimized to two significant figures (by trial and error) for each simulation as listed below. (a): $\Delta t = 2/450$, $2/1200$, $2/4300$ for $M_{x1} = 16$, 32, 64, respectively; (b): $\Delta t = 0.025$, 1.1 for $c = 2$, 100, respectively.

optimal preconditioner (7), (19) is faster then the non-preconditioned ITEM for all practical values of M_{x1}. (Taking M_{x1} to be under 10 grid points per one period of $V_{\rm ext}$ will make the non-preconditioned ITEM faster, but simulations with such a low resolution will yield a strongly distorted solution.)

V. SUMMARY OF UNSUCCESSFUL ATTEMPTS TO IMPROVE CONVERGENCE OF THE ME-ACCELERATED AITEM BY REDUCING ERROR OSCILLATIONS

- We tried to restart ME (i.e., stop it and return to it after various numbers of no-ME AITEM iterations), in analogy to restarting certain other methods [S7]. We varied the stopping criterion for ME (e.g., did it when the error would grow by a specified factor between 2 and 10, or ran ME only for a specified number of iterations). The oscillations did not, in general, reduce, and in some cases would stall at a higher error than the tolerance.

- When the error would continue to grow either for a specified number of iterations or by a given factor, we would flip the sign of parameter s to negative and kept it so for a (typically small) prescribed number of iterations. The error would decay in the few initial iterations but then would grow or stall.

- As explained in Sec. II B, the idea behind ME is that $Lu_{{\rm slow},j}$ is aligned with $L\phi_j$ so that ${\rm corr}[Lu_{{\rm slow},j}L\phi_j] \equiv \langle Lu_{{\rm slow},j}|L\phi_j\rangle/(\,\|Lu_{{\rm slow},j}\|\,\|L\phi_j\|\,) \approx -1$. Therefore, we monitored ${\rm corr}[Lu_{{\rm slow},j}L\phi_j]$ and flipped the sign of γ to make it positive when we detected that ${\rm corr}[Lu_{{\rm slow},j}L\phi_j] > 0$. This led to the same results as in the previous item.

- From Fig. 8(c) of the main text, one can see that the error grows when the smallest of the computed α's decreases. The decrease of α implies, via (8b), that γ becomes more negative. To arrest this behavior,

we tried to set the lower bound for γ in (37) at a less negative number, e.g., -10^{-5}. However, the error would continue to grow, but slower, and the "bulge" in the error curve would simply take many more iterations.

- Guided by the same observation about the correlation between the evolutions of α and the error, we tried to limit values of α to a range as follows: Once we computed $\alpha \equiv \alpha_{\text{first ME}}$ at the first ME iteration, we restricted α to a range $[\alpha_{\text{first ME}}/\text{sf}, [\alpha_{\text{first ME}} \cdot \text{sf}]$. Here sf is some factor, with whose value we also experimented. This did not consistently reduce either the duration or the magnitude of the error oscillations.

- In a slight variation to the idea of the previous item, we modulated the ME parameter s depending on the value of α at a given iteration; we explored various dependencies of s on α. The result was the same as in the previous item.

- As a variation of the previous idea, we modulated s according to the error rather than α. We tried various modification of:

$$s_{n+1} = s_n \, \frac{\|\text{error}_{n-1}\|}{\|\text{error}_n\|}.$$

  This sometimes reduced the duration and/or magnitude of the oscillation of the error evolution, but never by more than 20 % and not in all cases.

- In a different line of thought, once we detected that the error has grown by a specified factor, we would add a spatially smooth random perturbation to each atom, the size of the perturbation being on the order of the error at the given iteration. The hypothesis behind this was that when the error begins to grow, the iterations could be in the vicinity of some metastable state, and adding noise to the solution would push it away from that state and could restore convergence of the iterations. This did not work, and the error would typically stall.

- We also tried a number of variations of the stochastic gradient descent method. For example, we updated, by the AITEM equation (6a), only a subset of ϕ_j's. Such a subset, different for different iterations, was chosen either randomly or based on the size of the error of individual atoms (estimated either from $\|(\phi_j)_{n+1} - (\phi_j)_n\|$ or from $\left\|\left(L^{(0)}\phi_j\right)_n\right\|$. While the error of the updated atoms would diminish, it would grow back when a different subset of atoms would get updated later, and the overall result was that the average error would stall.

- Finally, we tried to include memory in the definitions of α_n and $\phi_{\text{slow},\,n}$, by replacing α_n with $(1 - \text{mf})\,\alpha_n + \text{mf}\,\alpha_{n-1}$, where α_n was computed by (20b) and mf was some 'memory factor' that we varied between 0 and 1. A similar update was used for $u_{\text{slow},\,n}$. This trick would occasionally help by a small amount, around 10%, and only for small mf $\lesssim 0.1$.

VI. PATTERNS FOR $ff = 7/16$ AND $7/12$

These patterns were reported by previous researchers, specifically: for $ff = 7/16$ — in Figs. 2(c) of [S8], 5(b) of [S9], and 2 of [S10]; for $ff = 7/12$ — in Figs. 5 of [S8], 7 of [S9], and 2(a) of [S11]. Our patterns, shown in Fig. S3, agree with those up to, possibly, a 90° rotation of the computational window.

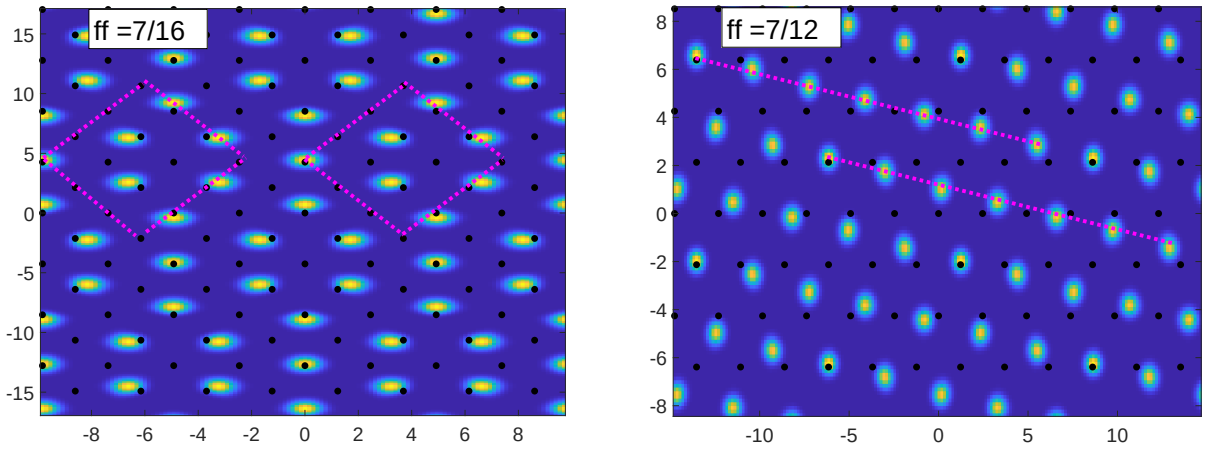

FIG. S3. Patterns for $ff = 7/16$ and $7/12$. See caption of Fig. 10 of the main text for explanations.

VII. QR-VERSION OF THE MME ALGORITHM

As noted in Appendix D, the condition number of matrix

$$P^{1/2}\widehat{\Delta}\,(P^{1/2}\widehat{\Delta})^T \tag{S18}$$

becomes very large for the number of eliminated modes $K \gtrsim 10$, which compromises the numerical calculation of its Cholesky decomposition (31) and hence the accuracy of the computed eigenmodes of $P^{-1}L$. One can avoid the computation of (S18) by first computing the row-counterpart of the QR-factorization of $P^{1/2}\widehat{\Delta}$:

$$P^{1/2}\widehat{\Delta} = \mathbf{R}\,\widehat{Q}, \tag{S19}$$

where $\mathbf{R}$ is a lower-triangular $K \times K$ matrix and $\widehat{Q}$ is the matrix of the same dimensions as $\widehat{\Delta}$ whose *rows* are orthonormal. Their computation is described below. Once such $\widehat{Q}$ and $L\widehat{Q}$ (defined below) are found, they can be used in place of $P^{1/2}\,\widehat{\Delta}$ and $P^{-1/2}\,L\widehat{\Delta}$ in the Algorithm of Sec. IV A. The condition number of $\widehat{Q}\widehat{Q}^T$ is 1 by design. The ill-conditioning of $\widehat{\Delta}$ is inherited by $\mathbf{R}$, which will need to be inverted (see below). However, $\mathrm{cond}\mathbf{R} \sim \mathrm{cond}\widehat{\Delta} = \sqrt{\mathrm{cond}\,\widehat{\Delta}\,\widehat{\Delta}^T}$, and thus the calculations involving $\widehat{Q}$ are much less ill-conditioned than those involving (S18).

We will now outline how (S19) is found. First, the rows of $P^{1/2}\widehat{\Delta}$ are orthonormalized by the modified Gram–Schmidt algorithm. The latter is used instead of the classical Gram–Schmidt because the rows of $\widehat{\Delta}$

are almost linearly dependent. The modified Gram–Schmidt algorithm of orthonormalizing K vectors $\vec{d}_i$ is the following (see, e.g., [S12]):

- Copy all $\vec{d}_i$ to $\vec{v}_i$.

- for $i = 1 : K$
  $r_{ii} = \|\vec{v}_i\|_2; \qquad \vec{q}_i = \vec{v}_i / r_{ii}$
  for $j = (i+1) : K$
  $r_{ij} = \vec{q}_i^{\,T} \vec{v}_j; \qquad \vec{v}_j = \vec{v}_j - r_{ij} \vec{q}_i$
  end
  end

The resulting vectors $\vec{q}_i$ are orthonormal. They form the rows of $\widehat{Q}$. The matrix $\mathbf{R}$ is constructed next.

One begins with constructing matrices

$$\widetilde{\mathbf{R}}_i = \mathbf{I}_i + \begin{pmatrix} \mathcal{O}_{(i-1),K} \\ \mathcal{O}_{1,K-i-1} \quad r_{ii} \ \ldots \ r_{iK} \\ \mathcal{O}_{(K-i),K} \end{pmatrix}, \tag{S20}$$

where $\mathbf{I}_i$ is the $K \times K$ identity matrix whose ith diagonal entry is replaced with 0, $\mathcal{O}_{j,l}$ is a zero matrix of size $j \times l$, and r_{ij} have been computed above. Then [S12]

$$\mathbf{R} = \left(\widetilde{\mathbf{R}}_K \, \widetilde{\mathbf{R}}_{K-1} \cdots \widetilde{\mathbf{R}}_1 \right)^T. \tag{S21}$$

Finally, to accomplish step (33) of the mME Algorithm, we need a counterpart of $L\widehat{\Delta}\,\widehat{\Delta}^T$ in terms of $\widehat{Q}$ and $L\,\widehat{Q}$. More specifically,

$$L\widehat{\Delta}\,\widehat{\Delta}^T \equiv P^{-1/2} L P^{-1/2} \; P^{1/2}\widehat{\Delta}\,(P^{1/2}\widehat{\Delta})^T \tag{S22}$$

(since all operators in the physical (i.e., (x, y)-) space commute with the boldfaced $K \times K$ matrices that act on rows of the $K \times (NM_xM_y)$ matrices with a hat). Therefore, we seek $P^{-1/2} L P^{-1/2} \; P^{1/2}\widehat{\Delta}$, which from (S19) becomes:

$$P^{-1/2} L P^{-1/2} \; P^{1/2}\widehat{\Delta} = \mathbf{R}\, P^{-1/2} L P^{-1/2}\, \widehat{Q}, \tag{S23a}$$

whence

$$P^{-1/2} L P^{-1/2}\, \widehat{Q} = \mathbf{R}^{-1}\, P^{-1/2} L \widehat{\Delta}. \tag{S23b}$$

To summarize: In this version of the mME Algorithm, one uses $\widehat{Q}$ instead of $P^{1/2}\widehat{\Delta}$, whence the r.h.s. of (31) is the identity matrix and hence $\mathbf{C} = \mathbf{I}$. In (33), $\mathbf{B}$ is computed as $P^{-1/2} L P^{-1/2}\, \widehat{Q}\, \widehat{Q}^T$.

The drawback of this version, as presented above, is that one would have to perform the (row-version of) QR factorization (S19) at every iteration, which has a relatively high operational cost of $2K^2\, O(NM_xM_y)$ [S12]. As an alternative, one may be able to design a variation of the modified Gram–Schmidt process where one of the vectors in the set (corresponding to the last row of $\widehat{\Delta}$) is discarded and a new row is made

orthogonal to all remaining $(K-1)$ rows. We did not explore this option since the more straightforward QR-version of the mME algorithm described above did not yield any consistent improvement in the iteration number compared to the version presented in the main text.

[S1] T. I. Lakoba, Math. Comput. Simul. **81**, 1572 (2011).
[S2] Here and below equations numbered without prefix 'S' refer to the main text.
[S3] W. Ryssens, M. Bender, and P.-H. Heenen, Eur. Phys. J. A **55**, 93 (2019).
[S4] L. Piela, Ideas of Quantum Chemistry (Elsevier, 2013) Chap. 8, 2nd ed.
[S5] K. Davies, H. Flocard, S. Krieger, and M. Weiss, Nucl. Phys. A **342**, 111 (1980).
[S6] J. McFarland and E. Manousakis, J. Phys.: Condens. Matter **33**, 055903 (2021).
[S7] B. O'Donoghue and E. Candès, Found. Comput. Math. **15**, 715 (2015).
[S8] P. Corboz, M. Boninsegni, L. Pollet, and M. Troyer, Phys. Rev. B **78**, 245414 (2008).
[S9] Y. Kwon and D. Ceperley, Phys. Rev. B **85**, 224501 (2012).
[S10] J. Happacher, P. Corboz, M. Boninsegni, and L. Pollet, Phys. Rev. B **87**, 094514 (2013).
[S11] J. Ahn, H. Lee, and Y. Kwon, Phys. Rev. B **93**, 064511 (2016).
[S12] L. Trefethen and D. Bau, Numerical linear algebra (SIAM, 1997) Chap. 8.