Journal of Computational Physics 531 (2025) 113921

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

Accelerating eigenvalue computation for nuclear structure calculations via perturbative corrections

Dong Min Roh [a,*], Dean Lee [b], Pieter Maris [c], Esmond Ng [a], James P. Vary [c], Chao Yang [a]

[a] Applied Mathematics and Computational Research Division, Lawrence Berkeley National Laboratory, Berkeley, CA 94720, United States of America

[b] Facility for Rare Isotope Beams and Department of Physics and Astronomy, Michigan State University, East Lansing, MI 48824, United States of America

[c] Department of Physics and Astronomy, Iowa State University, Ames, IA 50011, United States of America

ARTICLE INFO

Keywords:
Nuclear structure calculations
Eigenpair approximations
Rayleigh-Schrödinger perturbation theory
Subspace projection
Perturbative corrections

ABSTRACT

Subspace projection methods utilizing perturbative corrections have been proposed for computing the lowest few eigenvalues and corresponding eigenvectors of large Hamiltonian matrices. In this paper, we build upon these methods and introduce the term Subspace Projection with Perturbative Corrections (SPPC) method to refer to this approach. We tailor the SPPC for nuclear many-body Hamiltonians represented in a truncated configuration interaction subspace, i.e., the no-core shell model (NCSM). We use the hierarchical structure of the NCSM Hamiltonian to partition the Hamiltonian as the sum of two matrices. The first matrix corresponds to the Hamiltonian represented in a small configuration space, whereas the second is viewed as the perturbation to the first matrix. Eigenvalues and eigenvectors of the first matrix can be computed efficiently. Because of the split, perturbative corrections to the eigenvectors of the first matrix can be obtained efficiently from the solutions of a sequence of linear systems of equations defined in the small configuration space. These correction vectors can be combined with the approximate eigenvectors of the first matrix to construct a subspace from which more accurate approximations of the desired eigenpairs can be obtained. We show by numerical examples that the SPPC method can be more efficient than conventional iterative methods for solving large-scale eigenvalue problems such as the Lanczos, block Lanczos and the locally optimal block preconditioned conjugate gradient (LOBPCG) method. The method can also be combined with other methods to avoid convergence stagnation.

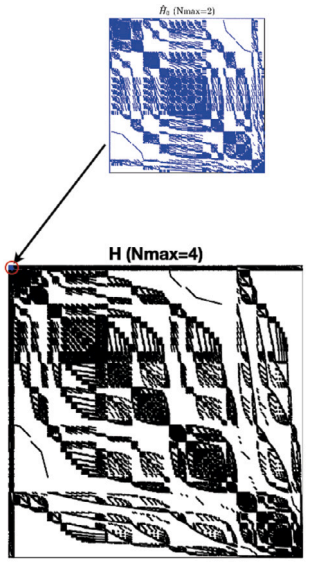
1. Introduction

Nuclear structure calculations require solving A-body Schrödinger equations where $A = Z + N$ is the number of nucleons consisting of Z protons and N neutrons. The Configuration Interaction (CI) method or no-core shell method (NCSM) [5], which represents the solution to the Schrödinger equation by a linear combination of A-body basis functions, reduces the problem to an algebraic eigenvalue problem:

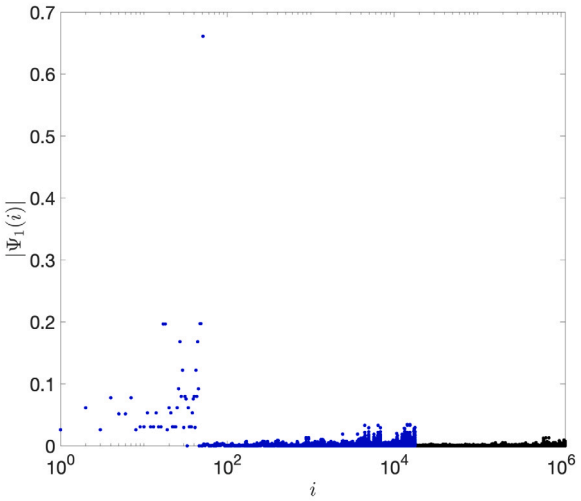
* Corresponding author.
E-mail address: droh@lbl.gov (D.M. Roh).

https://doi.org/10.1016/j.jcp.2025.113921
Received 2 August 2024; Received in revised form 13 January 2025; Accepted 6 March 2025

Available online 14 March 2025



(a) Sparsity structure of H and $\hat{H}_0$.



(b) Plot of the magnitudes of the components of the eigenvector Ψ_1 of H.

Fig. 1. (a) The Hamiltonian matrix H of ^{12}C constructed by the CI method with a truncation parameter $N_{\max} = 4$ and $\hbar\Omega = 20$ MeV, using the nucleon-nucleon interaction. The leading submatrix $\hat{H}_0$ (in blue) of H (in black) that is around 63 times smaller is equivalent to the Hamiltonian matrix constructed by the CI method with a truncation parameter $N_{\max} = 2$. (b) Illustration of the eigenvector localization observed in the Hamiltonian matrix. The leading components (in blue; corresponds to the portion of $\hat{H}_0$ to H) of the eigenvector corresponding to the lowest eigenvalue of H are several magnitudes larger than the tailing components. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

$$H\Psi_k = E_k\Psi_k, \tag{1}$$

where $H \in \mathbb{R}^{n\times n}$ is the matrix representation of the A-body nuclear Hamiltonian operator in a configuration space (spanned by a set of n A-body basis functions), E_k is the kth lowest eigenvalue of H representing an approximation to a discrete energy level, and Ψ_k is the corresponding eigenvector that contains the coefficients of the A-body basis functions in the expansion of the approximate eigenfunction in the A-body basis.

The dimension (n) of the matrix H depends on the number of nucleons A and the size of the CI model space (determined by a truncation parameter $N_{\max}$). Although n can be quite large, H is very sparse, and often only a few of its eigenpairs at the low end of the spectrum are of interest, making iterative methods suitable for solving (1).

The construction of the matrix H by the CI method is typically done in a hierarchical fashion where the leading submatrix of H corresponds to a matrix constructed from a smaller configuration space. Because the A-body basis functions that form the lower dimension CI space associated with a small $N_{\max}$ are typically more important than basis functions outside of such a configuration space, the eigenvectors of H tend to be localized; i.e., the leading components of the Ψ_k tend to be larger in magnitude, while the tailing components are relatively small in magnitude. We illustrate these properties of a nuclear Hamiltonian and its wavefunction using the nucleus ^{12}C as an example. The Hamiltonian is constructed with a nucleon-nucleon interaction Daejeon16 [36] where $\hbar\Omega = 20$ MeV describes the harmonic oscillator basis functions. Fig. 1(a) shows that the leading submatrix of H, which is around 63 times smaller, corresponds to a matrix constructed from a smaller configuration space. Fig. 1(b) shows that the eigenvector corresponding to the lowest eigenvalue is localized in a few low-lying components. The eigenvector localization suggests that the vector formed by padding the eigenvector of the leading submatrix with zeros can serve as a good initial guess for many iterative methods used to solve large-scale eigenvalue problems (1). Previous works [34,2] select such initial guesses for algorithms like the Lanczos algorithm [22], the block Lanczos algorithm [15], the Locally Optimal Block Preconditioned Conjugate Gradient (LOBPCG) algorithm [20], and the Residual Minimization Method with Direct Inversion of Iterative Subspace (RMM-DIIS) correction [40,19,32]. Another study [18] uses greedy algorithms to incrementally enlarge the submatrix and use the eigenvector of the enlarged submatrix as an improved starting guess to the Lanczos algorithm and the LOBPCG algorithm.

Subspace projection methods utilizing perturbative corrections have been previously proposed for solving large-scale eigenvalue problems [13,9]. In these methods, perturbative corrections to approximate eigenvectors are used to construct subspaces from which eigenpair approximations are obtained. However, the computational cost of these methods was not carefully analyzed and compared with those of state-of-the-art large-scale eigensolvers. In this paper, we build upon these ideas and present a detailed analysis and comparison of such methods with the state of the art conventional iterative eigensolvers. We introduce the term Subspace Projection with Perturbative Corrections (SPPC) to refer to this approach. Unlike previous work, we leverage the hierarchical structure of the NCSM Hamiltonian to define a partition of the Hamiltonian that allows efficient computation of the initial approximate eigenvectors and perturbative corrections.

In the SPPC method, the Hamiltonian matrix to be partially diagonalized is viewed as the sum of two matrices, i.e., $H = H_0 + V$, where the eigenpairs of unperturbed H_0 are relatively easy and inexpensive to compute, and V is viewed as a perturbation to H_0.

Perturbative corrections to eigenpairs of H_0 in successively higher order can be computed by solving a sequence of linear systems of equations. Together with the initial approximation to the desired eigenvector obtained from H_0, these correction vectors form a low dimensional subspace from which approximate eigenpairs of H are extracted through the Rayleigh-Ritz procedure [31]. When H_0 is chosen to be a block diagonal matrix diag($\hat{H}_0$,0), where $\hat{H}_0$ is the matrix representation of the A-body nuclear Hamiltonian in a smaller configuration space associated with a smaller $N_{\max}$ value, these linear systems can be solved efficiently in the small configuration space. The overall computational cost of the SPPC method grows linearly with respect to the highest order of perturbation included in the correction subspace. Adding each perturbative correction vector to the subspace and performing the Rayleigh-Ritz calculation requires multiplying the sparse matrix H with a single vector. We show numerically that the low dimensional subspace constructed in a low order SPPC method provides a more accurate approximation to a few lowest eigenvalues of H than a Krylov subspace of the same dimension constructed from the same starting guess. The method also appears to be more efficient than the locally optimal block preconditioned conjugate gradient (LOBPCG) method in early iterations even when a good preconditioner is available for the LOBPCG method. Although the SPPC method can stagnate as higher perturbative corrections are included, convergence stagnation can be mitigated by combining the SPPC with other iterative algorithms for solving large scale eigenvalue problems using the SPPC's eigenvector approximation as the starting guess for secondary algorithms.

The rest of this paper is organized as follows. In Section 2 we describe the basic Subspace Projection with Perturbative Corrections (SPPC) method for computing one eigenpair of H, and show how successively higher order perturbative corrections can be obtained from solutions of a set of linear systems of equations. We draw the connection between the SPPC and previously developed eigenvector continuation methods [12,33]. In Section 3, we present a version of the SPPC algorithm that can be used to compute a few lowest eigenpairs. A few practical implementation details of the SPPC method are discussed in Section 4. Numerical examples that demonstrate the efficiency of the SPPC method relative to other conventional large-scale eigenvalue computation methods are presented in Section 5. We also show the effectiveness of combining SPPC with a conventional eigensolver to avoid convergence stagnation.

2. Subspace projection with perturbative corrections (SPPC)

We split the matrix H into the sum of two matrices H_0 and V, i.e.,

$$H = H_0 + V, \tag{2}$$

where

$$H_0 := \begin{bmatrix} \hat{H}_0 & 0 \\ 0 & 0 \end{bmatrix}, \quad V := \begin{bmatrix} 0 & V_{12} \\ V_{21} & V_{22} \end{bmatrix}, \tag{3}$$

with the matrix $\hat{H}_0 \in \mathbb{R}^{n_0 \times n_0}$ (where $n_0 \ll n$) being the leading submatrix of H that corresponds to the representation of the Hamiltonian within a smaller configuration space.

The eigenvectors of H_0, which can be obtained from the eigenvectors of $\hat{H}_0$, are computed at a significantly lower computational cost compared to those of the full matrix H. They can be used as good initial guesses for conventional algorithms, such as the Lanczos and the LOBPCG algorithms, for computing the eigenpairs of H.

The SPPC method uses the eigenvectors of H_0 to initiate a subspace construction procedure to produce a subspace from which improved approximations of the desired eigenpairs can be obtained.

Instead of using the Lanczos algorithm or the LOBPCG method to construct the subspace, we use perturbative corrections to the initial eigenvector approximation to construct the subspace. In this approach, we introduce a scalar parameter c and define a family of perturbed matrices:

$$H(c) = H_0 + cV. \tag{4}$$

H can be viewed as the instance of the perturbed matrix (4) at $c = 1$.

We denote $E_k(c)$ and $\Psi_k(c)$ as the kth lowest eigenvalue of $H(c)$, assuming that no eigenvalue crossings occur within the parameter domain of interest. In practice, for the parameter ranges and the eigenvalues we consider, we do not encounter such crossings, and the ordering remains stable. It follows from the Rayleigh-Schrödinger perturbation theory [35] that these eigenvalues and eigenvectors can be written in terms of the perturbative corrections to the eigenvalues and eigenvectors of H_0, which we denote by $E_k^{(0)}$ and $\Psi_k^{(0)}$, i.e.,

$$\begin{cases} E_k(c) &= E_k^{(0)} + cE_k^{(1)} + c^2E_k^{(2)} + \cdots, \\ \Psi_k(c) &= \Psi_k^{(0)} + c\Psi_k^{(1)} + c^2\Psi_k^{(2)} + \cdots. \end{cases} \tag{5}$$

Here, $(E_k^{(p)}, \Psi_k^{(p)})_{p\geq 1}$ represent the pth order perturbative corrections to the eigenpair $(E_k^{(0)}, \Psi_k^{(0)})$, and are independent of the parameter c.

Substituting (5) into the equation (1) to obtain

$$H(c)\Psi_k(c) = E_k(c)\Psi_k(c) \tag{6}$$

and matching coefficients of the same degree yield the following set of equations

$$\begin{cases} E_k^{(p)} = (\Psi_k^{(p-1)})^T V \Psi_k^{(0)} \\ (H_0 - E_k^{(0)})\Psi_k^{(p)} = (E_k^{(1)} - V)\Psi_k^{(p-1)} + \sum_{l=0}^{p-2} E_k^{(p-l)} \Psi_k^{(l)} \end{cases} \tag{7}$$

that allow us to compute $E_k^{(p)}$ and $\Psi_k^{(p)}$ in a recursive fashion.

The asymptotic expansion used in (5) assumes that c is a small parameter. As a result, the expansion serves as a good approximation to the desired eigenpair only when c is sufficiently small, i.e., when c falls within the radius of convergence for (5), which is generally much smaller than 1. As a result, (5) cannot be used directly in general to approximate the kth eigenpair of H [13,10]. However, the eigenvector $\Psi_k^{(0)}$ of H_0 and the perturbative vectors $\Psi_k^{(p)}$ can be used to construct a subspace

$$\mathcal{M}_k^{(P)} := \text{span}\{\Psi_k^{(p)} : p = 0, 1, \dots, P\}, \tag{8}$$

from which approximation to E_k and Ψ_k can be obtained.

The idea of using perturbative corrections to construct an approximating subspace was proposed in [13,9] in the context of an eigenvector continuation (EC) method [12,33]. In an EC method, the aim is to approximate the eigenvectors of $H(c')$ at a new parameter value c' using information from known eigenvectors at other parameter values. Specifically, the eigenvectors of $H(c)$ for some choices of c's are used to construct a subspace from which approximations to the eigenvectors of $H(c')$ for $c' \neq c$ are obtained from the projection of $H(c')$ into such a subspace.

It was found in [13,9] that instead of using eigenvectors of $H(c)$ for several choices of $c \neq 1$ to construct a subspace from which approximate eigenvectors of $H(1)$ are extracted through the standard Rayleigh-Ritz procedure, more accurate approximations to the desired eigenpairs of $H(1)$ can be obtained from the subspace constructed from the eigenvector of H_0 as well as the perturbative eigenvector corrections as discussed above.

In a Rayleigh-Ritz procedure, we compute an orthonormal basis matrix $Q_k^{(P)}$ of $\mathcal{M}_k^{(P)}$ and form the projected matrix

$$\tilde{H} = \left(Q_k^{(P)}\right)^T H Q_k^{(P)}. \tag{9}$$

If (θ, q) is an eigenpair of $\tilde{H}$, then (θ, z) where $z = Q_k^{(P)} q$, yields an approximate eigenpair of H. We consider an approximate eigenpair to have converged if its relative residual norm

$$\frac{\|Hz - \theta z\|_2}{|\theta|} \tag{10}$$

is smaller than a preset tolerance value.

To obtain an approximate kth eigenpair $(\tilde{E}_k, \tilde{\Psi}_k)$ of H, we choose (θ, q) as the *lowest* eigenpair of $\tilde{H}$ in the above mentioned Rayleigh-Ritz procedure. Our choice of selecting the lowest eigenpair of the projected matrix is motivated by the Ky-Fan trace theorem [11], which states that the orthonormal eigenvectors Q associated with the lowest k_{ev} eigenvalues can be obtained by minimizing the trace operator of the projection, or equivalently, the sum of Rayleigh quotients under the orthonormality constraint $Q^T Q = I$:

$$\min_{Q^T Q = I_{ev}} \text{Tr}(Q^T H Q) = \min_{q_i^T q_j = \delta_{ij}} \left\{ q_1^T H q_1 + q_2^T H q_2 + \cdots + q_{ev}^T H q_{ev} \right\}. \tag{11}$$

In our approach, by selecting the lowest eigenpair of the projected matrix, we effectively minimize a Rayleigh quotient in (11). We present the SPPC method in Algorithm 1. We should point out that Algorithm 1 does not explicitly enforce the approximate eigenvectors to be orthonormal. However, if the selected eigenvectors of H_0 are sufficiently close to the desired eigenvectors of H, the orthonormality constraint can automatically be satisfied as the approximated eigenvectors converge to the desired eigenvectors.

Algorithm 1: The SPPC for the kth eigenpair.

Input: A nuclear CI Hamiltonian $H \in \mathbb{R}^{n \times n}$ partitioned as $H = H_0 + V$, where $H_0 = \text{diag}(\hat{H}_0, 0)$ with $\hat{H}_0$ constructed from a small configuration space (of dimension n_0); convergence tolerance (*tol*); and maximum order of perturbation allowed (*maxiter*)

Output: An approximate kth eigenpair $(\tilde{E}_k, \tilde{\Psi}_k)$ of H

Compute the kth nonzero eigenpair $(E_k^{(0)}, \Psi_k^{(0)})$ of H_0.
Set $\theta = (\Psi_k^{(0)})^T H \Psi_k^{(0)}$ and $z = \Psi_k^{(0)}$.
Return $(\tilde{E}_k = \theta, \tilde{\Psi}_k = z)$ if the relative residual norm (10) is less than *tol*.
for $p = 1, \dots,$ *maxiter* **do**
  Compute the correction energy $E_k^{(p)}$ and correction vector $\Psi_k^{(p)}$.
  Compute an orthonormal basis matrix $Q_k^{(p)}$ of $\mathcal{M}_k^{(p)}$ and form a projected matrix $\tilde{H}$.
  Compute the lowest eigenpair (θ, q) of $\tilde{H}$ and set $z = Q_k^{(p)} q$.
  Return $(\tilde{E}_k = \theta, \tilde{\Psi}_k = z)$ if the relative residual norm (10) is less than *tol*.
end

We should point out that the main distinction between the SPPC method proposed here and the EC method lies in the subspaces constructed in these methods and the cost of construction. Because EC projects H onto a subspace constructed from the eigenvectors of $H(c)$ for several (nonzero) c's, the cost of subspace construction may be just as expensive as the cost of solving the target eigenvalue

problem with a particular choice of c. On the other hand, because the subspace constructed in the SPPC method uses perturbative corrections that can be obtained by solving much smaller linear systems, the cost of subspace construction is significantly lower.

Although the basic idea of using perturbative corrections to construct a subspace for eigenvalue computations was presented in [13,9], the computational cost of this method was not carefully analyzed and compared with those of state-of-the-art large-scale eigensolvers. In [13], the eigenvectors of H_0 can be computed analytically for the one-dimensional quartic anharmonic oscillator. However, in general, identifying a H_0 that can be diagonalized analytically is not possible. In [9], the method is used to perform a A-body nuclear structure calculation. However, a different $H = H_0 + V$ splitting scheme is used, and the eigenvectors of H_0 are not easier to compute than those of H. In contrast, our splitting scheme results in an H_0 whose eigenvectors can be computed efficiently, as it corresponds to a smaller configuration space associated with a lower $N_{\max}$ value. In addition, our splitting allows us to construct perturbative corrections at a lower computational cost.

3. Targeting first few eigenpairs

Although we can use Algorithm 1 to compute each of the first k_{ev} eigenpairs one by one (or in parallel), approximations to larger eigenvalues and the corresponding eigenvectors appear to converge slowly, and sometimes to wrong values. It's not always the case that the lowest eigenpair of the projected matrix corresponds to the kth eigenpair of H. In some cases, the desired kth eigenpair may be the second lowest, third lowest, or even another eigenpair of the projected matrix. We show this in Section 6.

A more effective way to obtain approximations to the first k_{ev} eigenpairs is to combine the SPPC subspace $\mathcal{M}_k^{(P)}$ (8) constructed for each eigenpair to create a larger subspace

$$\mathcal{M}^{(P)} := \operatorname{span}\{\Psi_k^{(p)} : p = 0, 1, \dots, P, k = 1, \dots, k_{ev}\}, \tag{12}$$

from which k_{ev} approximate eigenpairs can be extracted simultaneously through the Rayleigh-Ritz procedure, by targeting the k_{ev} lowest eigenpairs of the projected matrix. Algorithm 2 shows how this approach works.

Algorithm 2: The SPPC for the first few eigenpairs.

Input: A nuclear CI Hamiltonian $H \in \mathbb{R}^{n\times n}$ partitioned as $H = H_0 + V$, where $H_0 = \operatorname{diag}(\hat{H}_0, 0)$ with $\hat{H}_0$ constructed from a small configuration space (of dimension n_0); number of desired eigenpairs (k_{ev}); convergence tolerance (*tol*); and maximum order of perturbation allowed (*maxiter*)

Output: Approximate k_{ev} lowest eigenpairs $\{(\tilde{E}_k, \tilde{\Psi}_k)\}_{k=1}^{k_{ev}}$ of H

Compute the eigenpairs $\{(E_k^{(0)}, \Psi_k^{(0)})\}_{k=1}^{k_{ev}}$ of H_0.
Compute an orthonormal basis matrix $Q^{(0)}$ of $\mathcal{M}^{(0)}$ and form the projected matrix $\tilde{H}$.
Compute the k_{ev} lowest eigenpairs $\{(\theta_k, q_k)\}_{k=1}^{k_{ev}}$ of $\tilde{H}$ and set $\{z_k = Q^{(0)} q_k\}_{k=1}^{k_{ev}}$.
Return $\{(\tilde{E}_k = \theta_k, \tilde{\Psi}_k = z_k)\}_{k=1}^{k_{ev}}$ if the relative residual norm (10) is less than *tol* for all $k = 1, \dots, k_{ev}$.
for $p = 1, \dots, maxiter$ **do**
    Compute the correction energy $E_k^{(p)}$ and correction vector $\Psi_k^{(p)}$ for $k = 1, \dots, k_{ev}$.
    Compute an orthonormal basis matrix $Q^{(p)}$ of $\mathcal{M}^{(p)}$ and form a projected matrix $\tilde{H}$.
    Compute the k_{ev} lowest eigenpairs $\{(\theta_k, q_k)\}_{k=1}^{k_{ev}}$ of $\tilde{H}$ and set $\{z_k = Q^{(p)} q_k\}_{k=1}^{k_{ev}}$.
    Return $\{(\tilde{E}_k = \theta_k, \tilde{\Psi}_k = z_k)\}_{k=1}^{k_{ev}}$ if the relative residual norm (10) is less than *tol* for all $k = 1, \dots, k_{ev}$.
end

4. Practical considerations

In this section, we describe a few practical implementation details of the SPPC algorithm.

4.1. Computing the correction vectors

We refer to the kth eigenvector of H_0, denoted by $\Psi_k^{(0)}$, as the zero-th order approximation to the kth eigenvector of H.

When H_0 is of the form given in (3), $\Psi_k^{(0)}$ can be obtained by computing the kth eigenvector of $\hat{H}_0$, denoted by $\hat{\Psi}_k^{(0)}$, and appending it with zeros to yield

$$\Psi_k^{(0)} = \begin{bmatrix} \hat{\Psi}_k^{(0)} \\ 0 \end{bmatrix}. \tag{13}$$

It follows from (7) and the block structures of H_0, V, and $\Psi_k^{(0)}$ that the first order corrections to the kth eigenvalue and eigenvector of H are

$$E_k^{(1)} = 0, \quad \Psi_k^{(1)} = \frac{1}{E_k^{(0)}} V \Psi_k^{(0)}. \tag{14}$$

It is easy to verify that

$$\Psi_k^{(1)} = \frac{H\Psi_k^{(0)} - E_k^{(0)}\Psi_k^{(0)}}{E_k^{(0)}},$$

i.e., the first order correction to the eigenvector is simply the residual associated with the zero-th order approximation.

Because $E_k^{(1)} = 0$, we can simplify the linear system in (7) to

$$(H_0 - E_k^{(0)})\Psi_k^{(p)} = -V\Psi_k^{(p-1)} + \sum_{l=0}^{p-2} E_k^{(p-l)}\Psi_k^{(l)}. \tag{15}$$

The matrix $H_0 - E_k^{(0)}$ is in a block diagonal form

$$H_0 - E_k^{(0)} = \begin{bmatrix} \hat{H}_0 - E_k^{(0)} & 0 \\ 0 & -E_k^{(0)} I \end{bmatrix} \tag{16}$$

consisting of two blocks where the first block $\hat{H}_0 - E_k^{(0)} \in \mathbb{R}^{n_0 \times n_0}$ is relatively small and the second block $-E_k^{(0)} I \in \mathbb{R}^{(n-n_0)\times(n-n_0)}$ is a scalar multiple of the identity matrix. As a result, solving the linear system in (15) essentially reduces to solving a much smaller linear system with $\hat{H}_0 - E_k^{(0)}$ being the coefficient matrix. If we partition $\Psi_k^{(p)}$ conformally with the blocks in (16) as $\Psi_k^{(p)} = [x_1, x_2]^T$ and the right-hand side of (15) as $b = [b_1, b_2]^T$ such that $x_1, b_1 \in \mathbb{R}^{n_0}$ and $x_2, b_2 \in \mathbb{R}^{n-n_0}$, x_2 can be easily computed as

$$x_2 = -\frac{1}{E_k^{(0)}} b_2, \tag{17}$$

and x_1 can be obtained by solving

$$(\hat{H}_0 - E_k^{(0)})x_1 = b_1. \tag{18}$$

Note that equation (15) is singular because $E_k^{(0)}$ is an eigenvalue of H_0. However, since $\mathcal{M}_k^{(P)}$ already includes $\Psi_k^{(0)}$ and we are only interested in contributions in the orthogonal complement of $\Psi_k^{(0)}$ from the solution of (15), we can project out $\Psi_k^{(0)}$ from the right-hand side of (15) before solving this equation. This is equivalent to projecting out $\hat{\Psi}_k^{(0)}$ from the right-hand side of (18), i.e. we solve

$$(\hat{H}_0 - E_k^{(0)})x_1 = (I - \hat{\Psi}_k^{(0)}(\hat{\Psi}_k^{(0)})^T)b_1. \tag{19}$$

4.2. Rayleigh-Ritz calculation

After obtaining the correction vectors, we generate an orthonormal basis matrix of the subspace spanned by these vectors and then perform the Rayleigh-Ritz procedure. We lay out these steps for approximating a single eigenpair and the first few eigenpairs.

Targeting the kth eigenpair. We use the Gram-Schmidt process [16] to obtain an orthonormal basis of the subspace $\mathcal{M}_k^{(p)}$. The orthonormal basis forms the columns of the matrix $Q_k^{(p)} \in \mathbb{R}^{n\times(p+1)}$. The $(p+1)$th column, denoted by $q_k^{(p)}$, is generated as follows.

$$\begin{aligned} \Phi_k^{(p)} &= \left[I - Q_k^{(p-1)}(Q_k^{(p-1)})^T\right]\Psi_k^{(p)}, \\ q_k^{(p)} &= \frac{\Phi_k^{(p)}}{\|\Phi_k^{(p)}\|_2}. \end{aligned} \tag{20}$$

We append $q_k^{(p)} \in \mathbb{R}^n$ to $Q_k^{(p-1)}$ such that

$$Q_k^{(p)} = [Q_k^{(p-1)}, q_k^{(p)}]. \tag{21}$$

Note that the projected matrix $(Q_k^{(p)})^T H Q_k^{(p)}$ can be constructed recursively. Assuming $(Q_k^{(p-1)})^T H Q_k^{(p-1)}$ has been computed in the previous step, we just need to compute $Hq_k^{(p)}$ and append an additional row and column to $(Q_k^{(p-1)})^T H Q_k^{(p-1)}$ as shown below.

$$\begin{bmatrix} (Q_k^{(p-1)})^T H Q_k^{(p-1)} & (Q_k^{(p-1)})^T H q_k^{(p)} \\ (q_k^{(p)})^T H Q_k^{(p-1)} & (q_k^{(p)})^T H q_k^{(p)} \end{bmatrix}. \tag{22}$$

Therefore, the major cost for constructing the projected matrix in each step of the SPPC method is in performing a single sparse matrix-vector multiplication (SpMV) in $Hq_k^{(p)}$.

Targeting the first few eigenpairs. To obtain an orthonormal basis $Q^{(p)} \in \mathbb{R}^{n\times k_{ev}(p+1)}$ for the combined subspace $\mathcal{M}^{(p)}$, we replace the Gram-Schmidt process (20) with the block Gram-Schmidt process [6]. If $\Psi^{(p)} = [\Psi_1^{(p)}, \Psi_2^{(p)}, \ldots, \Psi_{k_{ev}}^{(p)}]$, the block Gram-Schmidt procedure yields

$$\Phi^{(p)} = \left[I - Q^{(p-1)}(Q^{(p-1)})^T\right]\Psi^{(p)}. \tag{23}$$

We then perform a QR factorization of $\Phi^{(p)}$, i.e.,

$$\Phi^{(p)} = q^{(p)}R^{(p)}, \tag{24}$$

to generate an orthonormal basis $q^{(p)}$ for $\Phi^{(p)}$. We append $q^{(p)} \in \mathbb{R}^{n\times k_{ev}}$ to $Q^{(p-1)}$ such that

$$Q^{(p)} = [Q^{(p-1)}, q^{(p)}] \tag{25}$$

is an orthonormal basis for $\mathcal{M}^{(p)}$.

Again, the projected matrix $(Q^{(p)})^T H Q^{(p)}$ can be constructed recursively by utilizing the projected matrix from the previous step, computing $Hq^{(p)}$, and appending an additional row and column.

$$(Q^{(p)})^T H Q^{(p)} = \begin{bmatrix} (Q^{(p-1)})^T H Q^{(p-1)} & (Q^{(p-1)})^T H q^{(p)} \\ (q^{(p)})^T H Q^{(p-1)} & (q^{(p)})^T H q^{(p)} \end{bmatrix}. \tag{26}$$

The computational cost at this step is dominated by the cost for computing k_{ev} SpMVs in $Hq^{(p)}$.

4.3. Computational cost

We now discuss the overall computational cost of the SPPC method. We can see from Algorithm 1 and Algorithm 2 that the two major components of the SPPC algorithm are: (1) Solving the linear system (15); (2) forming the projected matrix $(Q^{(p)})^T H Q^{(p)}$. We have already shown that the projected matrix can be computed recursively using k_{ev} SpMVs in each step of the SPPC algorithm. The reduced linear system (18) of the correction equation (15) can be solved iteratively using, for example, the MINRES algorithm [30]. Because it has a much smaller dimension, the SpMVs performed in each MINRES iteration are relatively cheap. However, at the pth iteration, forming the right-hand side of the equation (15) requires multiplying V with $\Psi_k^{(p-1)}$ (for $p > 1$), which has nearly the same complexity as multiplying H with $\Psi_k^{(p-1)}$. Therefore, it may appear that each SPPC step requires performing $2k_{ev}$ SpMVs. We will show below that this is not the case. Both the right-hand side of (15) and the projected matrix can be obtained from the same $Hq^{(p-1)}$ product, which was computed in the previous iteration. As a result, each step of the SPPC algorithm only requires performing k_{ev} SpMVs.

Using the matrix splitting $H = H_0 + V$, we can rewrite $V\Psi^{(p-1)}$ as

$$V\Psi^{(p-1)} = H\Psi^{(p-1)} - H_0\Psi^{(p-1)} \tag{27}$$

where $\Psi^{(p-1)} = \left[\Psi_1^{(p-1)}, \ldots, \Psi_{k_{ev}}^{(p-1)}\right]$. Therefore, $V\Psi^{(p-1)}$ can be obtained by subtracting $H_0\Psi^{(p-1)}$, a much lower computational cost, from $H\Psi^{(p-1)}$.

We now show that $H\Psi^{(p-1)}$ can be easily obtained from $Hq^{(p-1)}$. It follows from (23) and (24) that

$$H\left[I - Q^{(p-2)}(Q^{(p-2)})^T\right]\Psi^{(p-1)} = Hq^{(p-1)}R^{(p-1)}. \tag{28}$$

As a result, we can obtain $H\Psi^{(p-1)}$ from $Hq^{(p-1)}$ by using the following identity

$$H\Psi^{(p-1)} = HQ^{(p-2)}(Q^{(p-2)})^T\Psi^{(p-1)} + Hq^{(p-1)}R^{(p-1)}. \tag{29}$$

Note that the analysis of the computational cost assumes $HQ^{(p-2)}$, which contains $Hq^{(j)}$ as its columns for $j = 0, 1, ..., p-2$, has been stored in memory.

As we will show in section 6, the highest order perturbation is often limited to 15, beyond which no significant improvement in the approximate eigenpair can be observed. Therefore, the dense linear algebra operations such as computing $(Q^{(p-2)})^T\Psi^{(p-1)}$ and diagonalizing the projected matrix can be performed with a relatively low cost compared to the cost of multiplying H with $q^{(p-1)}$.

5. Combining SPPC with other algorithms

As we will show in the next section, the perturbative correction is typically effective when the order of perturbation p is relatively low. The convergence of SPPC can stagnate when p increases, i.e., adding higher order perturbative correction may not help because they may be linearly dependent with respect to the basis vectors included in $\mathcal{M}^{(p)}$ already. In this case, it is useful to combine SPPC with another algorithm that can take the eigenvector approximation produced by SPPC as the starting guess.

Algorithms that can be combined with the SPPC method in a hybrid algorithm include, but are not limited, to the following:

Table 1
Properties of the test matrices H.

Nucleus	$N_{\max}$	dim(H)	dim($\hat{H}_0$)	nnz(H)	nnz($\hat{H}_0$)
^{6}Li	6	197,822	17,040	106,738,802	4,122,448
^{7}Li	6	663,527	48,917	421,938,629	14,664,723
^{11}B	4	814,092	16,097	389,033,682	2,977,735
^{12}C	4	1,118,926	17,725	555,151,572	3,365,009

- The Lanczos algorithm, which is a classical algorithm that generates an orthonormal basis of a Krylov subspace using the Gram-Schmidt procedure. It is initialized with an approximate eigenvector produced from the SPPC or a linear combination of k_{ev} approximate eigenvectors. It uses one SpMV per iteration to compute the next basis vector. The eigenpairs are approximated using the Ritz pairs obtained from the Rayleigh-Ritz procedure with the basis vectors of the Krylov subspace.
- The block Lanczos algorithm, which is a variation of the Lanczos algorithm that operates in blocks. It is initialized with a block of vectors approximating several eigenvectors of H and builds a Krylov subspace in blocks. For our numerical experiments, we initialize with k_{ev} eigenvectors and choose the block size of k_{ev}. This results in each iteration performing k_{ev} SpMVs. The main advantage of the block Lanczos algorithm over the Lanczos algorithm is that it can make use of approximations to several eigenvectors more effectively and most dense linear algebra operations can take advantage of level 3 BLAS. The computational efficiency of the block Lanczos algorithm is discussed in [17] and restart variants for overcoming the storage bottleneck are presented in [23,37,39,41].
- The LOBPCG algorithm, which is an iterative method that solves the equivalent trace minimization formulation of the eigenvalue problem. Similar to the block Lanczos algorithm, it can be initialized with approximations to several eigenvectors. For our numerical experiments, we initialize with k_{ev} eigenvectors, resulting in k_{ev} SpMVs for each iteration. One advantage of the LOBPCG algorithm is that it can utilize a preconditioner if it is available. The use of a good preconditioner can accelerate convergence [21,29,3]. A common choice for the preconditioner is a block diagonal part of the Hamiltonian matrix or a shifted matrix of the Hamiltonian for some appropriately chosen shift [25,20,2]. For our numerical experiments, we use a shifted preconditioner that involves a specific block diagonal part of the Hamiltonian matrix and a constant shift that approximates the lowest eigenvalue following the work in [2]. The LOBPCG algorithm has similar convergence properties as the Davidson [8,24,28] and Jacobi-Davidson [14,4] algorithms which we will not include in the discussions below.
- The residual minimization method (RMM) combined with the Direct Inversion of Iterative Subspace (DIIS) refinement algorithm (RMM-DIIS), which is a quasi-Newton algorithm for improving a specific eigenpair without computing other eigenpairs, provided that the initial approximation to the desired eigenpairs is sufficiently accurate. Each RMM-DIIS iteration performs one SpMV. The RMM-DIIS can also incorporate a preconditioner to accelerate convergence. For our numerical experiments, we choose the same preconditioner we use in the LOBPCG algorithm.

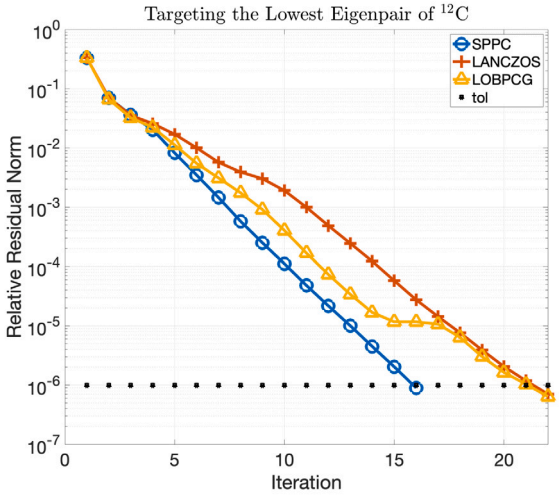
6. Numerical results

In this section, we demonstrate the effectiveness of the SPPC algorithm and compare it with other existing algorithms for computing the ground and a few low excited states of several light nuclei. We also show that the SPPC can be effectively combined with the RMM-DIIS to yield an efficient and accurate hybrid eigensolver for nuclear configuration interaction calculations. We call this hybrid eigensolver the SPPC+RMM-DIIS method.

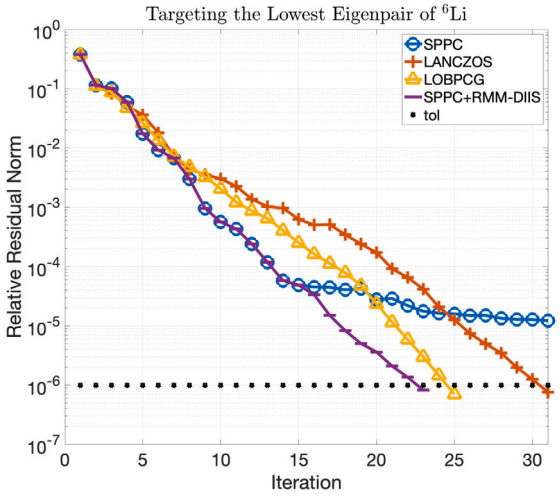
For all algorithms tested in this section, we use the relative residual norm (10) as the stopping criterion and set the convergence tolerance to be 10^{-6}. To ensure a fair comparison with the SPPC, we use the eigenvectors of the zero-order part H_0 as the initial guesses for the algorithms. All experiments were conducted using MATLAB.

6.1. Test matrices

We use the A-body Hamiltonian matrices corresponding to the nuclei ^{6}Li, ^{7}Li, ^{11}B, and ^{12}C in the following numerical experiments. The superscripts indicate the number of nucleons in the nuclei; for example, ^{7}Li indicates Lithium with 3 protons plus 4 neutrons. The Hamiltonian matrices H are constructed in a truncated CI space defined by a truncation parameter $N_{\max}$ and $\hbar\Omega = 20$ MeV, using the nucleon-nucleon interaction Daejeon16 [36]. For the same nucleus, a larger $N_{\max}$ results in a larger matrix H, but the size of the matrix is independent of the interaction. Note that with three-nucleon interactions, the number of nonzero matrix elements is an order of magnitude larger than the one for the nucleon-nucleon interactions, and the number of iterations and the actual eigenvalues of any eigensolver will be different. As we indicated earlier, the construction of H can be done in a hierarchical fashion so that a leading submatrix $\hat{H}_0$ of H corresponds to the same nuclear A-body Hamiltonian represented in a lower dimensional configuration space associated with a smaller $N_{\max}$. In this section, if H is the matrix representation of a nuclear A-body Hamiltonian represented in a configuration space associated with $N_{\max} = n_c$, the submatrix $\hat{H}_0$ corresponds to the representation of the same Hamiltonian in a configuration space associated with $N_{\max} = n_c - 2$. We list the dimension of H (denoted by dim(H)), the dimension of $\hat{H}_0$ (denoted by dim($\hat{H}_0$)), the number of non-zero elements in H (denoted by nnz(H)) and $\hat{H}_0$ (denoted by nnz($\hat{H}_0$)), and the $N_{\max}$ value associated with H in Table 1.



(a) Comparison of algorithms for computing the lowest eigenpair of ^{12}C.



(b) Comparison of algorithms for computing the lowest eigenpair of ^{6}Li.

Fig. 2. The convergence of algorithms for computing the lowest eigenpair of the Hamiltonians matrices (^{12}C on (a) and ^{6}Li on (b)). The major cost of one iteration equals one SpMV for all algorithms so that a direct comparison between the algorithms is possible in terms of the number of iterations needed for convergence.

6.2. *Targeting the lowest eigenpair*

We report the performance of the SPPC, the Lanczos algorithm, the LOBPCG algorithm, and the SPPC+RMM-DIIS for targeting the lowest eigenpair of the matrix H. As mentioned earlier, we use a preconditioner for the LOBPCG and the RMM-DIIS algorithms. The primary cost of all these algorithms is the number of SpMVs they perform before reaching convergence. Because each algorithm performs one SpMV per iteration, we can directly compare them by the number of iterations required to reach convergence.

The left plot of Fig. 2 shows the convergence history of the algorithms chosen for comparison for ^{12}C with respect to the iteration number. We observe that the SPPC algorithm converges in 16 iterations, which is the least among all methods, while the Lanczos algorithm and the LOBPCG algorithm converge in 22 iterations.

The result shown in the right plot of Fig. 2 is for the Hamiltonian associated with ^{6}Li. Several features of the SPPC algorithm are observed. The first observation is that the SPPC method converges more rapidly in the early iterations (up to the 15th iteration), and can be up to at most two orders of magnitude more accurate than other algorithms in these early iterations. However, the SPPC approximation appears to stagnate in subsequent iterations. This suggests that higher order correction vectors produced in later iterations (after iteration 15) do not contribute to improving the subspace constructed by the correction vectors produced in the early iterations. To verify this conjecture, we plot the angle between the current correction vector and the subspace spanned by the previous correction vectors, denoted by $\angle(\Psi^{(p)}, \mathcal{M}^{(p-1)})$, with respect to the iteration number p in Fig. 3. We observe that $\angle(\Psi^{(p)}, \mathcal{M}^{(p-1)})$ is relatively large in the first few SPPC iterations, and gradually decreases to the level of 10^{-5}. This is the point at which the new correction vector contributes minimally to the expansion of the subspace.

To overcome this stagnation, we consider a hybrid approach, the SPPC+RMM-DIIS, where we use the SPPC until the point of stagnation, and then switch to the RMM-DIIS. This hybrid approach takes advantage of the fast convergence of the SPPC for the first few iterations and the fast convergence of the RMM-DIIS when initialized with a good initial guess to the desired eigenvector. Specifically, we choose the initial guess in the RMM-DIIS as the Ritz vector produced from the SPPC method at the point of the switch. For ^{6}Li, we use the RMM-DIIS after the 15th iteration of the SPPC. We observe that the SPPC+RMM-DIIS breaks the stagnation of the SPPC and converges in 23 iterations, while the Lanczos algorithm and the LOBPCG algorithm converge in 31 and 25 iterations, respectively.

Fig. 4 shows the convergence history of the algorithms for the other two Hamiltonian matrices; ^{7}Li on the left plot and ^{11}B on the right plot. The SPPC stagnates at the order of magnitude 10^{-5} around the 15th iteration for the Hamiltonian ^{7}Li, while it stagnates at the order of magnitude slightly above 10^{-6} around the 17th iteration for the Hamiltonian ^{11}B. For these cases, we also consider the SPPC+RMM-DIIS. We observe that this hybrid approach converges the fastest with the fewest SpMVs performed. Table 2 provides a summary of the SpMV counts used by several algorithms tested in this section for all four Hamiltonian matrices.

6.3. *Targeting the five lowest eigenpairs*

We can use either Algorithm 1 or Algorithm 2 to compute the lowest k_{ev} eigenpairs. In the left plots of Fig. 5 and Fig. 6, we show the relative residual norms of the approximation to the 5 lowest eigenpairs of the ^{6}Li Hamiltonian at each iteration of Algorithm 1 and Algorithm 2, respectively. The relative residual norms associated with the first three eigenpairs drop below 10^{-4} by the 15th iteration. However, for the fourth and the fifth eigenpairs, the relative residuals obtained by Algorithm 1 jump to a larger value at a certain point and never become small in subsequent iterations. We can also observe that the Ritz values deviate from the true

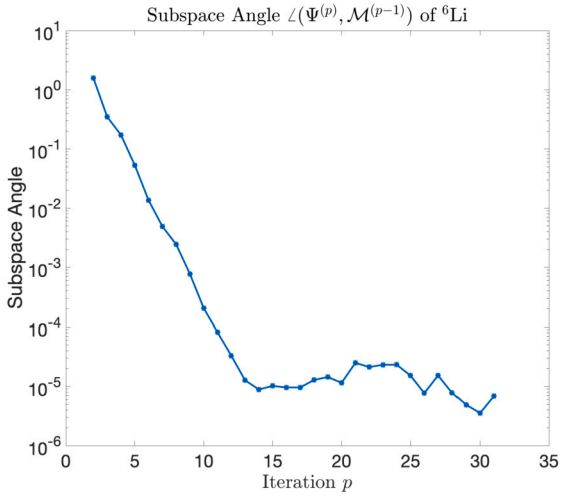
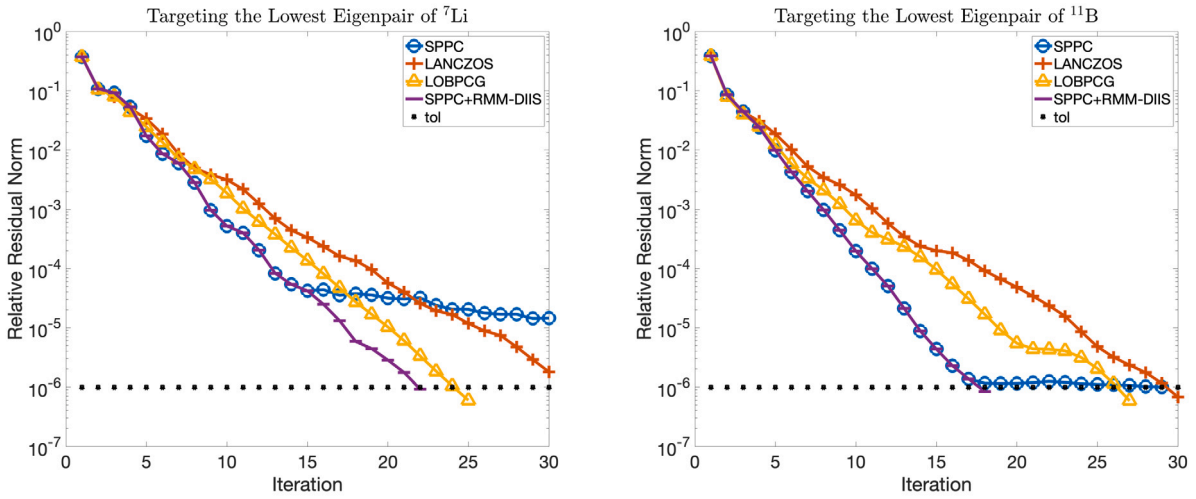


Fig. 3. The subspace angle between the correction vector at iteration p and the subspace spanned by the previous correction vectors from iteration 1 to $p-1$.

Table 2
SpMV count of the algorithms for computing the lowest eigenpair. The SPPC eventually converges for ^{11}B while it does not converge (DNC) for ^{6}Li, ^{7}Li. For these three Hamiltonians, the hybrid method, the SPPC+RMM-DIIS, is also considered where the RMM-DIIS switches with the SPPC.

Nucleus	SPPC	Lanczos	LOBPCG	SPPC+RMM-DIIS
^{6}Li	DNC	31	25	23
^{7}Li	DNC	31	25	22
^{11}B	29	30	27	18
^{12}C	16	22	22	

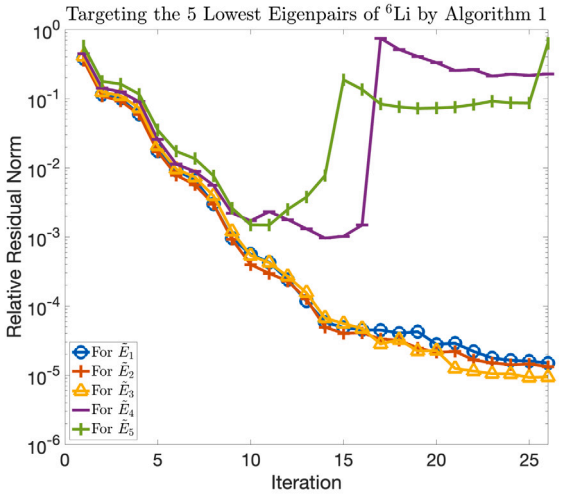


(a) Comparison of algorithms for computing the lowest eigenpair of ^{7}Li.

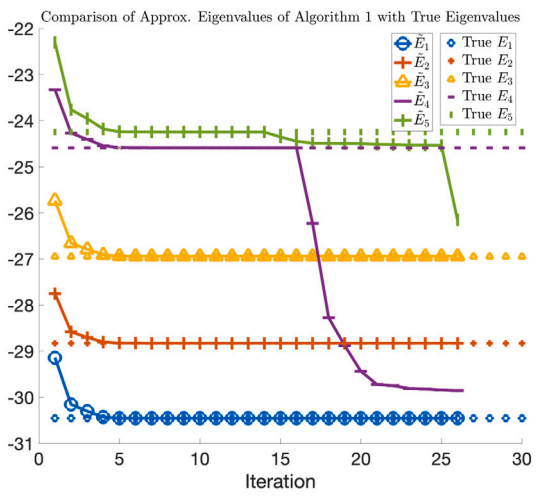
(b) Comparison of algorithms for computing the lowest eigenpair of ^{11}B.

Fig. 4. The convergence of algorithms for computing the lowest eigenpair of the Hamiltonians matrices (^{7}Li on (a) and ^{11}B on (b)). The major cost of one iteration equals one SpMV for all algorithms so that a direct comparison between the algorithms is possible in terms of the number of iterations needed for convergence.

eigenvalues of the Hamiltonian at the point of the jump, as shown in the right plot of Fig. 5. In these cases, the lowest eigenpair of the projected matrix no longer corresponds to the intended eigenvalue because the corrected eigenvector approximations were not mutually orthogonalized explicitly. Numerical round off error introduced in step 6 of Algorithm 1 moves the approximation to the fourth and fifth eigenpairs towards lower eigenpairs. When targeting the fourth eigenpair, the second lowest eigenpair of the projected matrix provides a better approximation, and when targeting the fifth eigenpair, it is the third lowest eigenpair that aligns with the

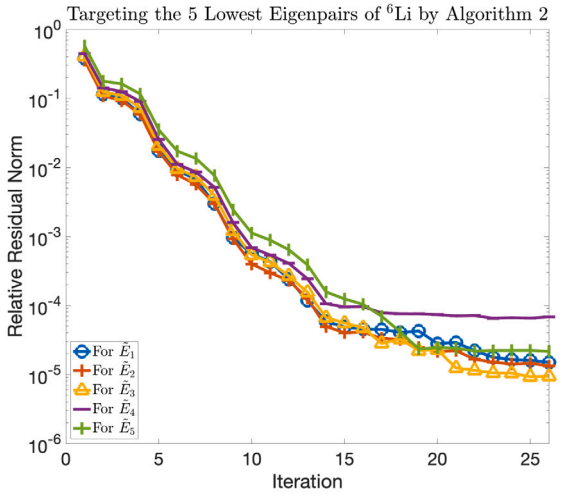


(a) Relative residual norms of the eigenpairs by Algorithm 1.

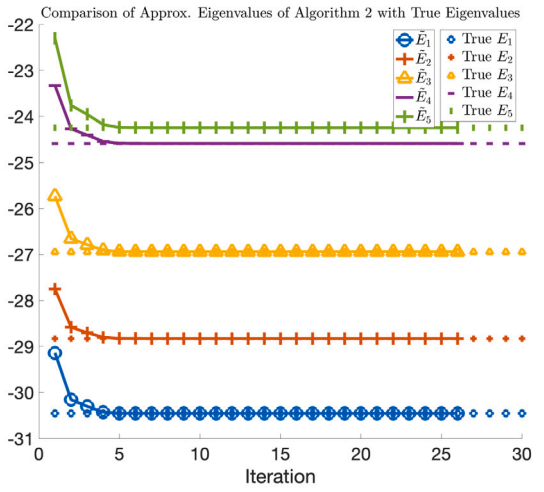


(b) Plot of the approximate eigenvalues of Algorithm 1, compared with the true eigenvalues.

Fig. 5. Algorithm 1 to compute the first 5 eigenpairs of the ^{6}Li Hamiltonian, one by one. (a) Relative residual norms for each eigenpair. (b) Comparison of approximated eigenpairs (solid lines) with the true eigenvalues (symbols without lines).



(a) Relative residual norms of the eigenpairs by Algorithm 2.



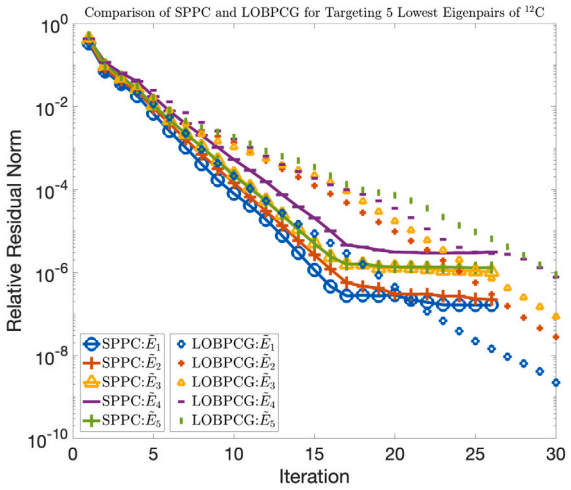
(b) Plot of the approximate eigenvalues of Algorithm 2, compared with the true eigenvalues.

Fig. 6. Algorithm 2 to compute the first 5 eigenpairs of the ^{6}Li Hamiltonian, one by one. (a) Relative residual norms for each eigenpair. (b) Comparison of approximated eigenpairs (solid lines) with the true eigenvalues (symbols without lines).

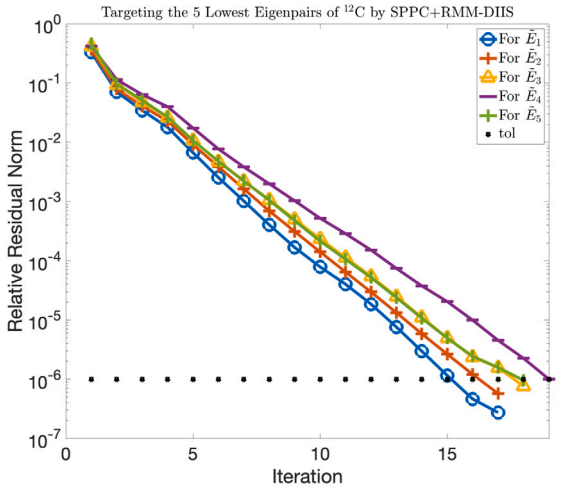
desired eigenpair. This discrepancy arises because the subspace constructed by Algorithm 1 for higher eigenpairs is not sufficient enough to isolate the target eigenvector. It is generally not possible to know a priori which of the eigenpairs of the projected matrix aligns with the target eigenpair. In contrast, as shown in the right plot of Fig. 6, by constructing a larger subspace consisting of perturbative corrections to several eigenvectors, Algorithm 2 computes approximate eigenvalues that do not deviate from the true eigenvalues. We observe in the left plot of Fig. 6 that all the approximate eigenpairs obtained by Algorithm 2 converge within a reasonable accuracy.

For the other three Hamiltonians ^{7}Li, ^{11}B, and ^{12}C, we observe a similar behavior as the ^{6}Li Hamiltonian where the relative residuals of the higher order eigenpairs obtained by Algorithm 1 would jump to a larger value at some point. Algorithm 2, on the other hand, has no such erratic behavior and refines eigenpair approximations for correct eigenpairs. Due to this observation, and because the driving cost of the two algorithms is both k_{ev} at each iteration, we use Algorithm 2 for approximating multiple lowest eigenpairs of the Hamiltonians.

When targeting a few lowest eigenpairs, the SPPC approximations can also stagnate in later iterations. In the left plot of Fig. 7, we observe that the relative residuals of the 5 lowest eigenpair approximations of the SPPC method for the ^{12}C Hamiltonian decrease



(a) Relative residual norms of the eigenpairs by the SPPC and the LOBPCG.



(b) Relative residual norms of the eigenpairs by the SPPC+RMM-DIIS.

Fig. 7. The convergence of algorithms for computing the 5 lowest eigenpairs of the ^{12}C Hamiltonian. (a) Convergence of the SPPC and the LOBPCG. The SPPC exhibits a more rapid convergence compared to the LOBPCG in the early iterations but starts to stagnate at the 17th iteration. (b) Convergence of the SPPC+RMM-DIIS. The SPPC+RMM-DIIS switches to the RMM-DIIS from the SPPC after the 17th iteration to escape the stagnation.

Table 3
SpMV count of the block Lanczos algorithm, the LOBPCG algorithm, and the SPPC+RMM-DIIS for computing the 5 lowest eigenpairs. The convergence of the SPPC tends to stagnate for some eigenpairs so the hybrid approach, the SPPC+RMM-DIIS, is considered. The RMM-DIIS switches from the SPPC after the 15th iteration for ^{6}Li and ^{7}Li, and after the 17th iteration for ^{11}B and ^{12}C.

Nucleus	Block Lanczos	LOBPCG	SPPC+RMM-DIIS
^{6}Li	155	140	119
^{7}Li	175	150	123
^{11}B	170	125	93
^{12}C	155	150	89

steadily until the 17th iteration, at which point the approximations stagnate. However, we also observe that the SPPC method converges more rapidly, up to at most two orders of magnitude accuracy, compared with the LOBPCG algorithm in these early iterations. This indicates that we can once again leverage the rapid convergence of the SPPC method in the early iterations and use the RMM-DIIS method to escape the stagnations when they occur. We switch to the RMM-DIIS method from the SPPC method when the relative residual norm of any of the eigenpairs starts to stagnate. In this hybrid approach, a separate RMM-DIIS run is used to refine each approximate eigenpair. It is initialized with the corresponding eigenvector approximation returned from the SPPC method at the point of the switch.

The right plot of Fig. 7 illustrates the convergence of the SPPC+RMM-DIIS for the ^{12}C Hamiltonian. We choose to switch to the RMM-DIIS from the SPPC at iteration 17, a point of stagnation for most eigenpairs. We observe that the SPPC+RMM-DIIS breaks the stagnation of the SPPC and that all 5 eigenpairs converge.

In Table 3, we present the SpMV counts of the block Lanczos algorithm, the LOBPCG algorithm, and the SPPC+RMM-DIIS for computing the 5 lowest eigenpairs of the four Hamiltonians. The switch to the RMM-DIIS algorithm occurs for the SPPC+RMM-DIIS after the 15th iteration for the ^{6}Li and ^{7}Li Hamiltonians, and after the 17th iteration for the ^{11}B and ^{12}C Hamiltonians. The block Lanczos algorithm and the LOBPCG algorithm are block methods, thus requiring continued iterations until every eigenpair converges and consequently a full k_{ev} SpMVs at each iteration as their driving cost. In contrast, the SPPC+RMM-DIIS targets each eigenpair individually after it switches to RMM-DIIS, so it does not incur more SpMVs for the converged eigenpair as it targets the non-converged eigenpairs. Due to this advantage and the leverage of the rapid convergence of the SPPC method in the early iterations, we observe that the SPPC+RMM-DIIS converges the fastest with the fewest SpMVs.

7. Conclusion

In conclusion, the Subspace Projection with Perturbative Corrections (SPPC) method, combined with the Residual Minimization Method with Direct Inversion of Iterative Subspace (RMM-DIIS), presents an advancement in the efficient computation of eigenpairs for large Hamiltonian matrices in nuclear structure calculations. The SPPC method leverages perturbative correction vectors

to enhance the accuracy of eigenpair approximations in the initial iterations, substantially reducing the number of sparse matrix-vector multiplications (SpMVs) required for convergence. Although the SPPC may experience stagnation in subsequent iterations, this challenge is effectively mitigated by integrating it with the RMM-DIIS algorithm, which provides robust refinement of eigenvector approximations. Our numerical experiments across several nuclear Hamiltonians demonstrate that the SPPC+RMM-DIIS hybrid approach outperforms traditional methods in terms of SpMVs. This hybrid method offers a promising solution for large-scale nuclear structure calculations, providing a reliable and efficient approach to solving the A-body Schrödinger equation.

While the preliminary results of the SPPC are promising, we have not yet provided a theoretical background explaining its effectiveness. We plan to address this in our future work, discussing the convergence behavior in detail. From a practical standpoint, we aim to develop a method to automatically detect stagnation, eliminating the need for manual decisions on when to switch to the RMM-DIIS. Additionally, we are interested in implementing the SPPC in a hybrid MPI/OpenMPI code, such as the software MFDn (Many-Fermion Dynamics for nuclear structure) [38,26,1,7,27], to be run at high-performance computing centers. We also want to explore further optimizations and applications of the SPPC to other large-scale eigenvalue problems in nuclear physics and beyond.

CRediT authorship contribution statement

Dong Min Roh: Writing – review & editing, Writing – original draft, Visualization, Validation, Methodology, Investigation, Conceptualization. **Dean Lee:** Writing – review & editing, Supervision, Project administration, Methodology, Funding acquisition, Conceptualization. **Pieter Maris:** Writing – review & editing, Supervision, Project administration, Methodology, Funding acquisition, Conceptualization. **Esmond Ng:** Writing – review & editing, Supervision, Project administration, Methodology, Funding acquisition, Conceptualization. **James P. Vary:** Writing – review & editing, Supervision, Project administration, Methodology, Funding acquisition, Conceptualization. **Chao Yang:** Writing – review & editing, Writing – original draft, Supervision, Project administration, Methodology, Investigation, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This material is based upon work supported by the Scientific Discovery through Advanced Computing (SciDAC) Program at the U.S. Department of Energyi (DOE), Office of Science under Grants DE-SC0023495 and DE-SC0023175 from the Office of Nuclear Physics, and also under funding for the FASTMath Institute from the Office of Advanced Scientific Computing Research through Contract No. DE-AC02-05CH11231. This work used resources at the National Energy Research Scientific Computing Center (NERSC), which is funded by DOE under Contract No. DE-AC02-05CH11231. In addition, D.L. acknowledges partial support from DOE grants DE-SC0024586, DE-SC0023658, DE-SC0013365, and NSF grant PHY-2310620. J.P.V. acknowledges partial support from DOE grants DE-SC0023692 and DE-0023707.

Data availability

Data will be made available on request.

References

[1] Hasan Metin Aktulga, Chao Yang, Esmond G. Ng, Pieter Maris, James P. Vary, Improving the scalability of a symmetric iterative eigensolver for multi-core platforms, Concurr. Comput., Pract. Exp. 26 (16) (2014) 2631–2651.
[2] Abdullah Alperen, Hasan Metin Aktulga, Pieter Maris, Chao Yang, Hybrid eigensolvers for nuclear configuration interaction calculations, Comput. Phys. Commun. 292 (2023) 108888.
[3] Merico E. Argentati, Andrew V. Knyazev, Klaus Neymeyr, Evgueni E. Ovtchinnikov, Ming Zhou, Convergence theory for preconditioned eigenvalue solvers in a nutshell, Found. Comput. Math. 17 (2017) 713–727.
[4] Zhaojun Bai, James Demmel, Jack Dongarra, Axel Ruhe, Henk van der Vorst, Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide, Society for Industrial and Applied Mathematics, Philadelphia, PA, 2000.
[5] Bruce R. Barrett, Petr Navrátil, James P. Vary, Ab initio no core shell model, Prog. Part. Nucl. Phys. 69 (2013) 131–181.
[6] Åke Björck, Numerics of Gram-Schmidt orthogonalization, Linear Algebra Appl. 197 (1994) 297–316.
[7] Brandon Cook, Patrick J. Fasano, Pieter Maris, Chao Yang, Dossay Oryspayev, Accelerating quantum many-body configuration interaction with directives, in: International Workshop on Accelerator Programming Using Directives, Springer, 2021, pp. 112–132.
[8] Ernest R. Davidson, The iterative calculation of a few of the lowest eigenvalues and corresponding eigenvectors of large real-symmetric matrices, J. Comput. Phys. 17 (1975) 87–94.
[9] Pepijn Demol, Thomas Duguet, Andreas Ekström, Mikael Frosini, Kai Hebeler, Sebastian König, Dean Lee, Achim Schwenk, Vittorio Somà, Alexander Tichai, Improved many-body expansions from eigenvector continuation, Phys. Rev. C 101 (4) (2020) 041302.
[10] Thomas Duguet, Andreas Ekström, Richard J. Furnstahl, Sebastian König, Dean Lee, Eigenvector continuation and projection-based emulators, arXiv preprint arXiv:2310.19419, 2023.
[11] Ky Fan, On a theorem of Weyl concerning eigenvalues of linear transformations I, Proc. Natl. Acad. Sci. USA 35 (11) (1949) 652.
[12] Dillon Frame, Rongzheng He, Ilse Ipsen, Daniel Lee, Dean Lee, Ermal Rrapaj, Eigenvector continuation with subspace learning, Phys. Rev. Lett. 121 (3) (2018) 032501.

[13] M. Companys Franzke, Alexander Tichai, Kai Hebeler, Achim Schwenk, Excited states from eigenvector continuation: the anharmonic oscillator, Phys. Lett. B 830 (2022) 137101.
[14] Gerard L.G. Sleijpen, Henk A. Van der Vorst, A Jacobi–Davidson iteration method for linear eigenvalue problems, SIAM Rev. 42 (2) (2000) 267–293.
[15] Gene H. Golub, Richard Underwood, The block Lanczos method for computing eigenvalues, in: Mathematical Software, Elsevier, 1977, pp. 361–377.
[16] Gene H. Golub, Charles F. Van Loan, Matrix Computations, JHU Press, 2013.
[17] Roger G. Grimes, John G. Lewis, Horst D. Simon, A shifted block Lanczos algorithm for solving sparse symmetric generalized eigenproblems, SIAM J. Matrix Anal. Appl. 15 (1) (1994) 228–272.
[18] Taylor M. Hernandez, Roel Van Beeumen, Mark A. Caprio, Chao Yang, A greedy algorithm for computing eigenvalues of a symmetric matrix with localized eigenvectors, Numer. Linear Algebra Appl. 28 (2) (2021) e2341.
[19] Zhongxiao Jia, G.W. Stewart, An analysis of the Rayleigh–Ritz method for approximating eigenspaces, Math. Comput. 70 (234) (2001) 637–647.
[20] Andrew V. Knyazev, Toward the optimal preconditioned eigensolver: locally optimal block preconditioned conjugate gradient method, SIAM J. Sci. Comput. 23 (2) (2001) 517–541.
[21] Andrew V. Knyazev, Klaus Neymeyr, Gradient flow approach to geometric convergence analysis of preconditioned eigensolvers, SIAM J. Matrix Anal. Appl. 31 (2) (2009) 621–628.
[22] Cornelius Lanczos, An iteration method for the solution of the eigenvalue problem of linear differential and integral operators, J. Res. Natl. Bur. Stand. 45 (1950) 255–282.
[23] R.B. Lehoucq, D.C. Sorensen, C. Yang, ARPACK Users' Guide: Solution of Large-Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1998.
[24] B. Liu, The simultaneous expansion method for the iterative solution of several of the lowest-lying eigenvalues and corresponding eigenvectors of large real-symmetric matrices, in: Numerical Algorithms in Chemistry: Algebraic Methods, 1978, pp. 49–53.
[25] Thomas A. Manteuffel, An incomplete factorization technique for positive definite linear systems, Math. Comput. 34 (150) (1980) 473–497.
[26] Pieter Maris, Masha Sosonkina, James P. Vary, Esmond Ng, Chao Yang, Scaling of ab-initio nuclear physics calculations on multicore computer architectures, Proc. Comput. Sci. 1 (1) (2010) 97–106.
[27] Pieter Maris, Chao Yang, Dossay Oryspayev, Brandon Cook, Accelerating an iterative eigensolver for nuclear structure configuration interaction calculations on GPUs using OpenACC, J. Comput. Sci. 59 (2022) 101554.
[28] Ronald B. Morgan, Davidson's method and preconditioning for generalized eigenvalue problems, J. Comput. Phys. 89 (1) (1990) 241–245.
[29] Klaus Neymeyr, A geometric convergence theory for the preconditioned steepest descent iteration, SIAM J. Numer. Anal. 50 (6) (2012) 3188–3207.
[30] Christopher C. Paige, Michael A. Saunders, Solution of sparse indefinite systems of linear equations, SIAM J. Numer. Anal. 12 (4) (1975) 617–629.
[31] Beresford N. Parlett, The Symmetric Eigenvalue Problem, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1998.
[32] Péter Pulay, Convergence acceleration of iterative sequences. The case of SCF iteration, Chem. Phys. Lett. 73 (2) (1980) 393–398.
[33] Avik Sarkar, Dean Lee, Convergence of eigenvector continuation, Phys. Rev. Lett. 126 (3) (2021) 032501.
[34] Meiyue Shao, H. Metin Aktulga, Chao Yang, Esmond G. Ng, Pieter Maris, James P. Vary, Accelerating nuclear configuration interaction calculations through a preconditioned block iterative eigensolver, Comput. Phys. Commun. 222 (2018) 1–13.
[35] I. Shavitt, R.J. Bartlett, Many-Body Methods in Chemistry and Physics: MBPT and Coupled-Cluster Theory, Cambridge University Press, Cambridge, UK, 2009.
[36] A.M. Shirokov, I.J. Shin, Y. Kim, M. Sosonkina, P. Maris, J.P. Vary, N3LO NN interaction adjusted to light nuclei in ab exitu approach, Phys. Lett. B 761 (2016) 87–91.
[37] Danny C. Sorensen, Implicitly restarted Arnoldi/Lanczos methods for large scale eigenvalue calculations, in: Parallel Numerical Algorithms, Springer, 1997, pp. 119–165.
[38] Philip Sternberg, Esmond G. Ng, Chao Yang, Pieter Maris, James P. Vary, Masha Sosonkina, Hung Viet Le, Accelerating configuration interaction calculations for nuclear structure, in: SC'08: Proceedings of the 2008 ACM/IEEE Conference on Supercomputing, IEEE, 2008, pp. 1–12.
[39] Gilbert W. Stewart, A Krylov–Schur algorithm for large eigenproblems, SIAM J. Matrix Anal. Appl. 23 (3) (2002) 601–614.
[40] D.M. Wood, Alex Zunger, A new method for diagonalising large matrices, J. Phys. A, Math. Gen. 18 (9) (1985) 1343.
[41] Yunkai Zhou, Yousef Saad, Block Krylov–Schur method for large symmetric eigenvalue problems, Numer. Algorithms 47 (4) (2008) 341–359.