A new workflow of shell-model calculations with the emulator and preprocessing using eigenvector continuation, and shell-model code ShellModel.jl

Sota Yoshida\textsuperscript{a,*}, Noritaka Shimizu\textsuperscript{b}

\textsuperscript{a}Liberal and General Education Center, Institute for Promotion of Higher Academic Education, Utsunomiya University, Mine, Utsunomiya, Tochigi 321-8505, Japan
\textsuperscript{b}Center for Nuclear Study, the University of Tokyo, Hongo, Bunkyo-ku, Tokyo 113-0033, Japan

Abstract

The configuration interaction method, which is well-known as the shell-model calculation in the nuclear physics community, plays a key role in elucidating various properties of nuclei. In general, these studies require a huge number of shell-model calculations to be repeated for parameter calibration and quantifying uncertainties. To reduce these computational costs, we propose a new workflow of shell-model calculations using a method called eigenvector continuation (EC). It enables us to efficiently approximate the eigenpairs under a given Hamiltonian by previously sampled eigenvectors. We demonstrate the validity of EC as an emulator of the shell-model calculations for a valence space, where the dimension of parameters is relatively large compared to the previous studies using EC. We also discuss its possible applications to the quantification of theoretical uncertainty, using an example of Markov chain Monte Carlo sampling for a simplified problem. Furthermore, we propose a new usage of EC: preprocessing, in which we start the Lanczos iterations from the approximate eigenvectors, and demonstrate that this can accelerate the shell-model calculations and the subsequent research cycles. With the aid of the eigenvector continuation, the eigenvectors obtained during the parameter optimization are not necessarily to be discarded, even if their eigenvalues are far from the experimental data. Those eigenvectors can become accumulated knowledge. In order to enable efficient sampling of shell-model results and to demonstrate the usefulness of the methodology described above, we developed a new shell-model code, ShellModel.jl. This code is written in Julia language and therefore flexible to add extensions for the users’ purposes.

Keywords: Shell model, Configuration interaction, Eigenvector continuation, Julia language

1. Introduction

The configuration interaction method plays a key role to study various properties of many-fermion systems. Nuclei are no exception; the nuclear shell model is one of the most powerful approaches to a microscopic understanding of static properties of nuclei \cite{chobanova2007,nachtmann2013}. Its application covers a wide range of the nuclear chart, and it also provides a bridge between \textit{ab initio} and phenomenological studies with the development in many-body methods to derive the shell-model Hamiltonians: Over the past decades, we have witnessed much progress in this direction with a nuclear force within chiral effective field theory (chiral EFT) \cite{chiralEFT1,chiralEFT2}. A representative and up-to-date example of the many-body method is the valence-space in-medium similarity renormalization group (VS-IMSRG) method \cite{vsimsrg1,vsimsrg2,vsimsrg3,vsimsrg4}, which has enlarged the reach of shell-model studies with the non-empirical interactions to nearly 700 isotopes \cite{vsimsrg5}.

In practice, however, it is still necessary to repeat a huge number of shell-model calculations to optimize the effective interactions, quantify uncertainties, extract physical intuitions by comparison with experimental observations, test the validity of many-body methods such as VS-IMSRG, and so on. The aim of this work is to propose a new workflow of shell-model calculations, i.e., how such a research procedure using shell-model calculations can be accelerated. The key ingredient is the eigenvector continuation, which was proposed in Ref. \cite{ec1}.

While much efforts have been spent on calibrating input parameters for nuclear models (e.g., low-energy constants in chiral EFT and phenomenological shell-model interactions), one still needs expensive numerical samplings in a high-dimensional parameter space to find a reasonable range of those parameters and to quantify the associated uncertainties in parameters and the target observables for a deeper understanding of properties of nuclei.

Recently, the importance of the uncertainty quantification (UQ) has been widely pointed out in various contexts, such as parameter calibration of the chiral EFT potentials \cite{chiralUQ1,chiralUQ2,chiralUQ3} and nuclear observables \cite{nukeft1,nukeft2,nukeft3,nukeft4}. This also applies to the shell model, but the UQ for the valence shell-model studies are limited up to the \textit{sd}-shell \cite{valenceUQ1,valenceUQ2}. This is due to the rapid growth in the size of matrices to be diagonalized as the number of valence nucleons or the size of model space (called valence space too) increases. To reduce the computational cost, the approximation method using the principal component analysis was proposed in Ref. \cite{valenceUQ2}. However, one still needs additional efforts and/or efficient methods for parameter calibration and UQ to enlarge the scope to a heavier or neutron/proton-rich

\textsuperscript{*}Corresponding author.

E-mail address: syoshida@cc.utsunomiya-u.ac.jp

Preprint submitted to Elsevier

May 19, 2021
2. Basics and features of ShellModel.jl

2.1. Nuclear shell model and its implementation

In shell-model calculations, the problem of interest is to solve the Schrödinger equation under a given effective interaction for valence nucleons, which is dependent on some parameter \( \tilde{c} \):

\[
H(\tilde{c})|\psi(\tilde{c})\rangle = E(\tilde{c})|\psi(\tilde{c})\rangle. \tag{1}
\]

The typical parametrization of the shell-model Hamiltonian is the following:

\[
H = H^{(1)} + H^{(2)} = \sum_{ac} \delta_{ac} c^\dagger_a c_c + \frac{1}{4} \sum_{abcd} h^{(2)}_{abcd} c^\dagger_a c^\dagger_b c_d c_c, \tag{2}
\]

where \( c^\dagger_a \) and \( c_c \) denote a creation and annihilation operator on the single-particle state \( a \) and \( c \), respectively. One can naturally extend the above to include the three-body (and higher many-body) term, but we will not go into the details of three-body forces in this work.

The eigenvectors, called wave functions too, are expressed as a superposition of many-body configurations represented by Slater determinants. As in other shell-model codes like KSHELL, ShellModel.jl employs the so-called \( M \)-scheme basis to express the wave function. This means that the single particle states, \( a, b, c, d \) in Eq. (2), are specified by its harmonic oscillator quanta, \( n, l, j \) (and the \( z \) component of angular momentum \( m \) and of isospin \( t_z \), if needed). Under this, the two-body part of shell-model Hamiltonian, \( H^{(2)} \), can be rewritten in a more explicit form:

\[
H^{(2)} = \frac{1}{4} \sum_{abcdJM} N_{ab}(J) N_{cd}(J) A^{\dagger}(ab; JM) A(cd; JM) V_f(abcd),
\tag{3}
\]

\[
N_{ab}(J) = [(1 + \delta_{ab})]^{1/2}, N_{cd}(J) = [(1 + \delta_{cd})]^{1/2},
\tag{4}
\]

\[
A^{\dagger}(ab; JM) = \sum_{m_a,m_b,m_c,m_d} (j_{am_a} j_{bm_b} J) e^{\dagger}_{j_{am_a}} J e_{j_{bm_b}} J M
\tag{5}
\]

\[
A(cd; JM) = \sum_{m_c,m_d} (j_{cm_c} j_{dm_d} J)(c_{j_{cm_c}} J c_{j_{dm_d}} J M,
\tag{6}
\]

where the label of single particle states, \( a,b,c,d \), denote the quanta \( n, l, j, t_z \), \( (j_{am_a} j_{bm_b} J) \), \( (j_{cm_c} j_{dm_d} J) \) are Clebsch-Gordan coefficients, and \( J, M \) are coupled angular momentum and its \( z \) component. Due to the symmetries of the nuclear force, the number of non-zero \( V_f(abcd) \) is typically on the order of tens or thousands, and can be further reduced by a factor of about two when assuming the isospin symmetry of the effective interaction.

Usually, the diagonal part of \( h^{(1)} (c = a) \) is called single-particle energies (SPEs), and \( V_f(abcd) \) in Eq. (3) is called two-body matrix elements (TBMEs). These SPEs and TBMEs are given in the interaction file for shell-model codes. Note that, in some codes, the so-called isospin formalism \( V_f(abcd) \) is used for interactions with the isospin symmetry. The typical recipe to determine these SPEs and TBMEs is the following: (i) making an initial guess in some way, (G-matrix, VS-IMSRG, etc.), and (ii) modifications of the parameters so as to minimize the chi-square deviation from the selected experimental data. There have been many previous works adopting the above recipe [36, 37, 38, 39, 40, 41]. One of the most successful, well-known examples is the USDB interaction for the \( sd \)-shell nuclei [40]. The USDB gives the root-mean-square deviation 126 keV for 380 energy data in the mass region \( A = 16 - 40 \).

Once the interaction is given and the model space is specified, the main task of shell-model codes is diagonalizing the shell-model Hamiltonian under the basis states, which is now in the \( M \)-scheme. The matrices are in general very sparse, but the total number of non-zero matrix elements is too large to be stored on memory and diagonalize. For this reason, the diagonalization is typically done by means of the Lanczos method.

In the rest of this subsection, we briefly explain the feature of ShellModel.jl. Most of the functions in ShellModel.jl is originally implemented combining other Julia packages, but we referred to MSHELL64 and KSHELL for implementation of some methods. More precisely, we implemented the following: the Thick-Restart Lanczos method, block family of the Lanczos methods, and the double Lanczos method [30, 42] for the projection to a specified total angular momentum \( J \). While the KSHELL code adopts the so-called on-the-fly generation of the matrix elements for the matrix-vector product, ShellModel.jl stores matrix elements rather explicitly on memory to balance the speed and moderate memory usage. The proton-neutron interaction is stored in the form of “one-body jumps” [24, 25]. The current version of ShellModel.jl is optimized to run on a relatively smaller environment such as a laptop. For smaller
systems like the sd-shell nuclei, ShellModel.jl is the fastest code (to the authors’ knowledge).

2.2. Eigenvector continuation: an efficient emulator and a preprocessing for the nuclear shell model

The eigenvector continuation (EC), which was proposed in Ref. [12], has been widely used as an efficient emulator of nuclear many-body methods [43]–[45] [46]–[47] and as a re-summation method [48]–[49]. In the following, we give a brief overview of the eigenvector continuation, and then explain how to apply it to shell-model calculations.

Suppose that the eigenvectors for Eq. (1) have been already obtained at $N_i$ different parameters $|\psi(\vec{r}_i^1), |\psi(\vec{r}_i^2)\rangle, \ldots, |\psi(\vec{r}_i^{N_i})\rangle$. These eigenvectors will be hereafter referred to as samples. In such a case, the eigenpairs under a given Hamiltonian $H(\vec{r}_0)$ are approximated by solving the following generalized eigenvalue problem in the subspace spanned by the samples:

$$H = \lambda N \vec{v},$$
$$\tilde{H}_{ij} = (\psi(\vec{r}_i^j)|H(\vec{r}_0)|\psi(\vec{r}_j^i),$$
$$N_{ij} = (\psi(\vec{r}_i^j)|\psi(\vec{r}_j^i)).$$

Then, the original eigenpairs can be approximated as

$$E(\vec{r}_0) = \lambda,$$
$$|\psi(\vec{r}_0)^i\rangle \approx \sum_{i=1}^{N_i} v_i|\tilde{\psi}(\vec{r}_i^j)\rangle \equiv |\tilde{\psi}_EC(\vec{r}_0)^i\rangle. \quad (11)$$

The problem is reduced from the diagonalization of a sparse Hamiltonian matrix $H$ with size of $M$-scheme dimension to the $H$ of a dense matrix of size $N_{ij}$, which is typically on the order of tens or hundreds. This might significantly reduce the computational cost compared to the original problem. Note that it is straightforward to extend Eqs. (7)–(11) to include excited states. With the approximated wave functions, one can approximate the expectation values for a target observable:

$$\langle \tilde{\mathcal{O}} \rangle \approx \langle \tilde{\psi}_EC(\vec{r}_0)|\tilde{\mathcal{O}}|\tilde{\psi}_EC(\vec{r}_0)\rangle, \quad (12)$$

where the operator $\mathcal{O}$ can be e.g., electromagnetic transition operators.

The eigenvector continuation has been well known as the Rayleigh-Ritz method in (applied) mathematics, and introduced to or re-evaluated in the nuclear physics community recently. However, its properties are still under consideration [50], and the previous works in nuclear physics have proved for the first time its efficiency for large-scale many-body problems and enlightened the possibility to enlarge the scope of microscopic studies to a heavier and/or more exotic region of the nuclear landscape.

For solving the EC problems, the most time-consuming part is, in general, to evaluate the expectation value of Hamiltonians, i.e., $\tilde{H}$ in Eq. (8). However, the computational cost for evaluating $\tilde{H}$ can be somewhat alleviated in shell-model calculations. This is because of the fact that the expectation values of shell-model Hamiltonians can be factorized out by the SPEs and TBMEs:

$$\tilde{H}_{ij} = \sum_k h_k^{(1)} \times \text{OBTD}_k + \sum_k v_k^{(abcd)} \times \text{TBDT}_k, \quad (13)$$

where $k$ denotes the labels of one- and two-body interactions in a given shell-model Hamiltonian $H(\vec{r}_0)$, and the OBTD and TBTD are the abbreviations of one- and two-body transition densities, respectively. The concrete expressions for the OBTDs and TBTDs are summarized in Appendix A. Once we could evaluate the transition densities in Eq. (13) for arbitrary two sample eigenvectors, one can significantly reduce the computational cost to evaluate $\tilde{H}$ for different parameters, because these can be evaluated simply by the dot product of the parameters and the transition densities, which are independent on the parameter values.

Although the number of required samples depends on the problem and desired accuracy, the cost for solving generalized eigenvalue problem Eq. (7) is typically negligible compared to that of the original eigenvalue problem. Hence, the eigenvector continuation provides an efficient emulator of the shell-model calculation as well as other nuclear models discussed in the previous works.

In addition to its role as an emulator, EC can also be used as a preprocessing method. In the Lanczos method, one usually starts with the Lanczos iterations from a random vector, when one has no prior information about the starting vector. However, the convergence of the Lanczos iterations should be accel-
erated by starting from a good initial guess of the eigenvectors. The approximate eigenvectors constructed by EC would make it, and it will be also helpful to increase the sample size for the EC emulator.

As will be detailed in the next section, we propose a new workflow of the shell-model calculations, which is schematically shown in Fig. 1. The eigenvector continuation can be used (a) as an efficient emulator, (b) as a preprocessing to accelerate the exact calculations, and (c) to generate another sample eigenvector efficiently with the help of the preprocessing. We expect that the proposed method would form acceleration cycles of the shell-model calculations and subsequent research workflows.

3. Results

In this section, we present the main results of this study. For demonstration purposes, we restrict ourselves to consider the sd shell on top of $^{19}$O core, i.e., the model space consists of $1s_{1/2}, 0d_{3/2}, 0d_{5/2}$ orbits (12 single-particle states in total) for both protons and neutrons. Since we consider only the positive parity states, the plus sign on the total angular momentum $J$, e.g., $0^+, 2^+$, will be omitted where appropriate.

In Sec. 3.1 we briefly describe our procedures to prepare the sample eigenvectors. Next, the efficiency of the EC emulator is demonstrated in Sec. 3.2 and Sec. 3.3. We also make some remarks on the accuracy of approximated eigenvectors by evaluating the magnetic dipole moments and electric quadrupole moments. Finally, in Sec. 3.5 we present the results of the preprocessing of the Lanczos method using the EC approximate wave functions and discuss its possible accelerating effect on convergence.

3.1. Calculations of sd-shell nuclei with ShellModel.jl

The new shell-model code, ShellModel.jl, is designed to make sample eigenvectors efficiently and to demonstrate the efficiency of the emulator and the preprocessing with EC. With ShellModel.jl, the execution time to evaluate 10 lowest states of $^{28}$Si, which has the largest $M$-scheme dimension in the full sd-shell space (93,710 for $M = 0$), is about 3 seconds on a MacBook Air (2020, Apple M1). We provide the sample codes in the GitHub repository [35] and brief instructions in Appendix B for the sample codes.

To make the sample eigenvectors, we generated 50 random sd-shell interactions by adding gaussian random values with the standard deviation $\sigma_{\text{int}} = 1.0$ MeV to the USDB interaction [36]. The USDB is the phenomenological sd-shell interaction constructed by $G$-matrix [51] and the chi-square fit using 608 data in 77 nuclei. Hereafter, the terms “random interaction” and “sample” denote those by $\sigma_{\text{int}} = 1.0$ unless otherwise mentioned. In accordance with the USDB interaction having the isospin symmetry, the Hamiltonians are defined in the 66-dimensional parameter space (3 for SPEs and 63 for TBMEs). Next, we calculated the five lowest eigenvectors with $J = 0, 1, 2, 3, 4$ (for even nuclei) and $J = 1/2, 3/2, 5/2, 7/2, 9/2$ (for odd) under the 50 random interactions, i.e. sampling 250 eigenvectors in total for each $J$.

3.2. Approximate eigenpairs by the eigenvector continuation

In what follows, we show the results of approximate eigenpairs by EC. Using the samples discussed in the previous subsection, we solved Eq. (7) to estimate the approximate eigenvalues for other 100 random interactions, which were made in the same way as above, and compare them to the exact values.

In Fig. 2 we show the EC estimates of five yrast states for $^{28}$Si ($J = 0, 1, 2, 3, 4$) and $^{25}$Mg ($J = 1/2, 3/2, 5/2, 7/2, 9/2$) in comparison with the exact values. One can see from Fig. 2 that the symbols are on the diagonal (dotted line) while the absolute values of energy under the 100 random interactions spread over the relatively large range. As shown in Table 1, the typical size of the relative errors to the exact values is less than 1% in energy, when all the $N_s = 250$ samples for each $J$ is used. Here we chose the four nuclei in the middle of sd shell ($^{28}$Si, $^{26}$Al, $^{25}$Mg, and $^{24}$Mg) as examples. We found that the relative errors for the odd-odd nucleus, $^{26}$Al, are worse than those for the others.

To show the dependence on the sampling procedure, we summarized the results for five different $N_s$ in Table 1. The sample size $N_s$ is the product of the number of random interactions and the number of excited states used as the sample eigenvectors for EC, and some cases are marked with an asterisk to indicate that the detailed conditions differ from the others.
the $N_e = 50^*$ case means that only the first and second lowest states under the first half of the 50 random interactions are used, while the $N_e = 250^*$ case corresponds to the result using the samples with the standard deviation for random interactions $\sigma_{\text{int}}$ increased from 1.0 to 3.0. The eigenvalues of the five yrast states of $^{28}\text{Si}$ under the 50 random interactions with $\sigma_{\text{int}} = 3.0$ spread over a wide range of $-350$ to $-50 \text{ MeV}$, but the relative errors are still $\sim 3\%$. This indicates that one can make use of the eigenvector continuation for rough parameter optimization even if the prior knowledge of the parameter range is poor.

From these results, one can see that there are two strategies to improve the accuracy of the EC estimates. One is to sample more excited states; the $N_e = 250$ case using the five lowest states as the samples improves the accuracy by a factor of two in terms of the relative errors from the $N_e = 50$ case. The other one is to increase the sample points in the parameter space. Looking at the results of the $N_e = 50$ and $N_e = 50^*$ cases, we expect that the latter one is more beneficial unless we change the number of states of interest. On the other hand, the former one is relatively easy in terms of computational costs. Since the efficiency may depend on the sampling procedure and the distribution of eigenvalues, how to select the strategy and how many states to be sampled will be determined by the trade-off between the costs and desired accuracy. Note that one can receive benefit from the factorization of transition densities in Eq. (13) during the sampling procedure; to increase the samples, all one needs is the evaluation of transition densities among already sampled eigenvectors and the new sample.

The execution time to calculate approximate eigenvalues for a random $sd$-shell interaction is about a few tens of milliseconds in our settings. It significantly facilitates the sampling procedures for the parameter optimization or uncertainty quantification.

### 3.3. Uncertainty quantification

In this subsection, we consider the Monte Carlo sampling over the parameter space utilizing the EC emulator. The following analysis is by no means an exhaustive study of quantifying uncertainties in the $sd$-shell effective interactions, but it is meant to illustrate the usefulness of the emulator.

We simplify the problem by picking up twelve energy data of $^{28}\text{Si}$ from those used to construct the USDB [36]: $0^+_2, 1^+_2, 2^+_1, 3^+_1, 4^+_1, 1^+_3, 2^+_3, 3^+_3, 4^+_3$. This is apparently too small data set to constrain the 66 parameters in the $sd$-shell interaction (in the isospin formalism), but enough for the demonstration purpose.

For this small data set, we consider to get samples obeying a certain posterior distribution. Let the log-likelihood be

$$
\log L(\vec{c}) = -\frac{1}{N} \sum_{i=1}^{N} \frac{(E_{\text{EC}}(\vec{c}) - E_{\text{Exp},i})^2}{2\sigma_{\text{err},i}^2},
$$

where $E_{\text{EC}}(\vec{c}), E_{\text{Exp},i}$, and $\sigma_{\text{err},i}$ are the EC estimate of energy, its corresponding experimental value, and the expected typical error of the calculation. Since the errors stemming from EC are dominant, we fixed the $\sigma_{\text{err},i}$ as follows

$$
\sigma_{\text{err},i}^2 = \sigma_{\text{EC,typ}}^2 + \sigma_{\text{EC},i}^2,
$$

where the first term is fixed as 1.0 MeV, which is typical error in order of 1% of the energy eigenvalues, and the second term is small correction with $i$ dependence, which is taken from the difference of $E_{\text{EC}}$ with $N_e = 250$ under the USDB interaction from the corresponding exact value. Since the number of data in our data set is less than the number of parameter, we introduced the independent Gaussian prior, which is equivalent to the L2 regularization, to avoid highly multi-modal structure of the parameter space. The log-prior is defined as

$$
\log Pr(\vec{c}) = -\frac{\Lambda}{2} ||H(\vec{c}) - H(\vec{c}_{\text{ref}})||_2^2.
$$

Here $|| \cdot ||_2$ is the L2 norm, i.e., the sum of squared deviation between a parameter $\vec{c}$ and the reference value $\vec{c}_{\text{ref}}$, which is now the USDB. The choice of variance of the Gaussian prior $\Lambda$ is non-trivial, but we fixed $\Lambda = 10$ to simplify the discussions.

To sample from the posterior $\propto L(\vec{c}) \times Pr(\vec{c})$, we adopted a Markov chain Monte Carlo (MCMC) method with the Metropolis-Hastings algorithm. We set a random interaction around the USDB parameters as an initial state of the MCMC, and generated 100,000 MCMC samples (after 10,000 burn-in) with the Metropolis-Hastings updates. The elapsed time to achieve it with the EC emulator on a laptop was about 30 minutes. In Fig. 3 we show one realization of parameter-distributions for some selected channels. The posteriors are drawn by the histograms with a bin width of 125 keV. The dot symbols and dashed lines denote the USDB and the prior distributions by independent normal distributions. The complete plot including other TBMEs is shown in Fig. C.6. As shown in Appendix C we have confirmed that the independent runs give the converged energy distributions.

The posterior is similar to the prior, but some are slightly modified through the log-likelihood, Eq. (14). This indicates that the USDB values for these channels were affected and thereby constrained by a variety of data, not just the $^{28}\text{Si}$ data with lower $J$. In this way, the efficient emulator allows a detailed sensitivity analysis on the parameter and adopted data. There still remain a lot of investigations undone such as dependence on the choices of the loss function (log-likelihood), experimental data, and $\sigma_{\text{err},i}$, and the more efficient sampling schemes for MCMC. In addition to these, it is also desired to...
Figure 3: Probability density distributions for some selected parameters. The x-axis is the parameter value in units of MeV, and the y-axis is probability density for

develop a method to quantify the uncertainties taking into account of the variational property of the EC estimates for the energy eigenvalues. However, the above analysis manifests that one can achieve the samplings, which were computationally too heavy to carry out on a laptop, in tens of minutes with the help of the EC emulator. It is encouraging to proceed towards full evaluation of uncertainties for deeper understanding of the relation among nuclear structure, effective interactions, the choice of the experimental data, and the underlying nuclear force.

3.4. Accuracy of the approximate wave functions

Next, we test the accuracy of the approximate wave functions by evaluating other observables using Eq. (12). In Fig. 4 we show the approximated values of magnetic dipole moments $\mu$ and electric quadrupole moments $Q$ for the yrst states of the four nuclei in the middle of sd shell. The symbols denote the yrst states with $J$ ranging 1/2, 1, ..., 9/2 which are allowed by the selection rules, i.e., $J \neq 0$ for $\mu$ and $J \neq 0, 1/2$ for $Q$. One can see that the agreements for both $\mu$ and $Q$ become worse than that for energy eigenvalues. This is simply because the electromagnetic transitions are much sensitive to the accuracy of the wave functions.

Note, however, that the correspondence between the exact and approximated eigenstates are deduced simply from their energy eigenvalues. As described below, it is highly non-trivial to find one by one correspondence among them. It should be also noted that the values are given in units of nuclear magneton $\mu_N = 0.105$ e$^2$/fm$^2$ for $\mu$ and $e$ fm$^2$ for $Q$, and we used the free-nucleon values for the $g$-factors ($g_N = 1.0$, $g_m = 0$, $g_{np} = 5.586$, and $g_{np} = -3.826$) and effective charges ($e_p = 1.5, e_n = 0.5$), while these choices do not have much effect on the correlation plot.

Most of the approximate values of the magnetic dipole moment are nearly on the diagonal line in Fig. 4 (a). The points for even nuclei with $J = 1, 2, 3, 4$ concentrate on $\mu \sim 0.5, 1.0, 1.5, 2.0$, respectively. On the other hand, results of electric quadrupole moments are rather scattered, while most of the results are still on the diagonal line. As an extreme example, let us look at the lower right-most triangle in Fig. 4 (b), which is $J = 4^+$ state with $Q_{\text{Exact}} = +28.340$ (e fm$^2$). Diagonalizing the corresponding random interaction, we found that the first and second eigenstates have small energy difference $\sim 400$ keV and, the $Q$-moments with opposite sign and similar amplitude. While the approximate energy eigenvalues are close to the exact ones with $\sim 2\%$ accuracy, the $Q$ moment by EC has the opposite sign to the exact ones:

$$E_{\text{Exact}}(4^+) = -75.951 \text{ MeV, } Q_{\text{Exact}}(4^+) = +28.340 \text{ efm}^2,$$

$$E_{\text{Exact}}(4^+) = -75.454 \text{ MeV, } Q_{\text{Exact}}(4^+) = -25.682 \text{ efm}^2,$$

$$E_{\text{EC}}(4^+) = -74.751 \text{ MeV, } Q_{\text{EC}}(4^+) = -25.635 \text{ efm}^2,$$

$$E_{\text{EC}}(4^+) = -73.825 \text{ MeV, } Q_{\text{EC}}(4^+) = +27.599 \text{ efm}^2. \quad (17)$$

This indicates that the approximate eigenvectors for the nearly degenerate states can be in the wrong order in terms of the energy, or contaminated by other states.

In addition to the energy eigenvalues, the moments and electromagnetic transitions are much useful for sensitivity analyses on the input parameters and on adopted approximations and truncations (if there were). When discussing these observables, one needs additional efforts to improve the accuracy of the approximate wave functions.

3.5. Acceleration of the Lanczos iterations by preprocessing

From the results shown in the previous subsections, it is expected that the convergence of the EC estimates to the exact values is not so fast as a function of the sample size. While the eigenvector continuation provides an efficient emulator, one still needs exact calculations to discuss the observables which demand high accuracy of the wave functions.

In general, the matrix-vector product of the shell-model Hamiltonian matrix takes up most of the calculation time, and one usually starts the Lanczos iterations from a normalized random vector. In the following, we consider the preprocessing, which is to start the Lanczos iterations from the EC approximate eigenvectors, to accelerate the shell-model calculations. For simplicity, we discuss the numbers of Hamiltonian operations of the Lanczos iterations with and without the preprocessing, although the elapsed time of the computation depends not only on the number of the operations but also on various conditions such as computing environment and the block size for a block algorithm.

As an example, we consider the four $sd$-shell nuclei discussed above, and fix the target shell-model Hamiltonian to the USD$B$ interaction. In Table $\text{C}_{6}$ we summarized the number of iterations, which is equivalent to the number of Hamiltonian operations, required to get converged results of the lowest states with the specified $J$. The number of iterations for the block
Lanczos method is summarized in Table 3. When considering the excited states too, we use the block algorithm for the Lanczos method and both the block size \( q \) and the number of states of interest \( n \) are fixed as four to simplify the discussions. It means that we prepare the four lowest approximate eigenvectors from the samples and then use them as the starting block vectors with the size of \( q = 4 \).

As a general trend, we can see that the number of iterations becomes smaller as the number of samples increases. This tendency is prominent in the case of the single target state, Table 2. One exception is seen in the \( {}^{26}\text{Al} \) in Table 3, the \( N_t = 50^* \) and \( N_t = 150 \) cases are worse than the \( N_t = 50 \) case. This may indicate that in order to speed up the convergence, the initial guesses of the nearly degenerate states must be made simultaneously with high accuracy. Indeed, the exact eigenvalues of third and fourth \( J = 0 \) states of \( {}^{26}\text{Al} \) are close to each other with a difference of about 50 keV.

It should be noted that the preprocessing can deteriorate the convergence with some naive usage. If one calculates the ten lowest \( J = 0 \) states of \( {}^{24}\text{Mg} \) starting from four approximate eigenvectors which are constructed from the \( N_t = 250 \) (50 \( \times \) 5) samples, one may need additional several iterations compared to a random starting vector.

The convergence patterns of the block Lanczos iterations are shown in Fig. 4. The four columns from the top to the bottom corresponds to the case of \((q,n) = (4,4), (10,4), (4,10), (10,10), \) respectively. Here, the \( q \) is the block size and the \( n \) is the number of states of interest. The right panels are difference in a logarithmic scale from the exact value. Note that the tolerance for the Lanczos method is set as \( 10^{-6} \), and the difference smaller than the tolerance is regarded as \( \log_{10}(E_{\text{Exact}} - E_{\text{Ritz}}) = -6 \) in the log-plot. The solid line with circles and the dashed lines with diamonds correspond to the cases of preprocessed and random initial vectors, respectively. The symbols represent the points at which the results converged.

As already discussed above, the \((q,n) = (4,4) \) case exhibits the effect of the preprocessing. Looking at \((q,n) = (10,4) \), we can see that the number of Hamiltonian operation is much reduced. On the other hand, the \((q,n) = (4,10) \) case shows that insufficient preprocessing can deteriorate the convergence, and the effect of the preprocessing is limited for the \((q,n) = (10,10) \) case due to the ninth and tenth lowest states. This is because the \( N_t = 250 \) (50 \( \times \) 5) samples have less information about the higher excited states.

As shown in Appendix D, the efficiency of the preprocessing depends on the problem. A rule of thumb to receive the benefit of the preprocessing is starting with a larger number of states than that of interest. It does not always reduce the computation time, because the number of manipulations in each Lanczos iteration becomes larger as the block size increases. In such a case, it would be better to start with a large initial block vec-

---

**Figure 4**: (a) The EC estimates of magnetic dipole moments \( \mu \) in units of nuclear magneton \( \mu_N = 0.105 \text{ fm}^2 \). (b) The quadrupole moments \( Q \) in units of \( \text{e fm}^2 \) in comparison to the exact values. The symbols denote the yrast states with \( J \) ranging from 1/2 to 9/2 by increment of 1/2 which are allowed by the selection rules. The values are evaluated with the free-nucleon values for \( g \)-factors and effective charges \( (e_p = 1.5, e_n = 0.5) \).
4. Summary and Outlook

We demonstrated that the eigenvector continuation can be used as an efficient emulator for shell-model calculations, as in previous works for other models [43-48]. As an example, we considered sd-shell nuclei and the randomly sampled effective interactions around the USDB interaction.

With the sample eigenvectors under the 50 random interactions, we demonstrated that the exact eigenenergies under other 100 random interactions in a similar range can be estimated with accuracy of a few percent. The EC emulator is found to be helpful for rough parameter optimization and quantifying uncertainty. The latter was partially demonstrated by MCMC sampling on a simplified problem.

It should be noted that one should pay attention to the accuracy of the approximate wave functions by EC when discussing the observables such as moments and electromagnetic transitions. In such a case, one needs additional efforts to improve the accuracy of the approximate wave functions. As one solution to make it feasible, we proposed a new usage of the eigenvector continuation as the preprocessing for the shell-model calculations. The exact diagonalization and subsequent manipulations can be accelerated by starting the Lanczos iterations from the EC approximate eigenvectors.

As is schematically shown in Fig. [1] the eigenvectors in the optimization process of effective interactions are no longer necessarily to be discarded, even if their eigenvalues are far from the corresponding experimental data. These eigenvectors are not wasted, but can become accumulated knowledge and accelerate the workflow of researchers.

In addition to the methodology, we developed the new opensource shell-model code, ShellModel.jl, written in the Julia language. The code enables us to sample shell-model results efficiently, and is highly flexible to add any extensions to suit the users’ purposes such as MCMC samplings. The main codes and sample scripts are provided in the GitHub repository.

There are many possible future works along this line. While the microscopically derived effective interactions give reasonable results, it is still indispensable to construct better phenomenological interactions for more quantitative discussions in comparison with experimental data, designing new experiments, and testing the validity of many-body methods (and the approximations in them) such as VS-IMSRG. Although it is still challenging, the methods described in this work would pave the way for finding better phenomenological effective interactions for larger systems, e.g., a universal pf-shell interaction.

For parameter calibration using the EC emulators, one may need a more efficient sampling strategy especially when the computational cost for one sample is high. It would be interesting to apply an experimental design using Bayesian optimization. For uncertainty quantification, the efficient emulator may allow to evaluate the full posterior for the parameters and/or observables using Markov chain Monte Carlo methods. This would contribute to a long journey towards a better understandings of how nuclei emerge from the underlying nuclear force.

It is also desirable to open a database of transition densities and wave functions so that one can easily obtain approximate eigenpairs and benefit from the preprocessing without repeating massive computations. This will further facilitate the cycle of cross-validation between theory and experiment.

5. Acknowledgments

This work was partly supported by KAKENHI grants (17K05433) from JSPS, “Priority Issue on post-K computer” (Elucidation of the Fundamental Laws and Evolution of the Universe), and “Program for Promoting Researches on the Supercomputer Fugaku” (simulation for basic science: from fundamental laws of particles to creation of nuclei), MEXT, Japan. We also thank Multidisciplinary Cooperative Research Program by Center for Computational Sciences, Tsukuba University (xg18i035).
Appendix A. Transition densities

It is convenient to introduce the so-called transition densities. In this appendix, we summarize our definition of the transition densities in Eq. (13) assuming that the shell-model wave functions are given in the $M$-scheme, $|\psi_{JM}\rangle$.

The one-body transition densities (OBTDs) is given as

$$\text{OBTD}(fi; J_aJ_b, J) \equiv \frac{1}{\sqrt{2J + 1}} \langle \psi_{JM} | [c_j^\dagger \otimes \tilde{e}_{jb}]^J | \psi_{JM} \rangle,$$

(A.1)

where we introduced $\tilde{e}_{jb} \equiv (1)^{J_m} e_{jb}$, and $[\cdot \otimes \cdot]^J$ means taking the so-called reduced matrix element, and the notation $[\cdot \otimes \cdot]^J$ is for the rank-$J$ irreducible tensor operators. For more details on the tensor algebra, see e.g., [52, 53]. Since we are interested in the $\lambda = 0$ scalar in terms of irreducible tensor operator and the diagonal $(f = i, J_f = J_i, M_f = M_i)$ component, which contributes to $H$ in Eq. (13), $\text{OBTD}$ for the $k$-th single particle state is defined as

$$\text{OBTD}_k \equiv \sqrt{\frac{2J + 1}{2I + 1}} \text{OBTD}(ii; jkJ; 0) = \langle \psi_{JM} | N_k | \psi_{JM} \rangle,$$

(A.2)

where $N_k$ is the occupation number of the $k$-th orbital, and the factor $\sqrt{\frac{2J + 1}{2I + 1}}$ is introduced to make $\text{OBTD}$ identical with the occupation number of $k$-th orbital.

The two-body transition densities (TBTDs) are defined as

$$\text{TBTD}(f; abcd; J_{ab}J_{cd}; \lambda) \equiv \frac{1}{\sqrt{2J + 1}} \langle \psi_{JM} | [A^\dagger (ab; J_{ab}M_{ab}) \otimes \tilde{A}(cd; J_{cd}M_{cd})]^J | \psi_{JM} \rangle,$$

(A.3)

$$\text{TBTD}(f; abcd; J_{ab}J_{cd}; \lambda) \equiv \frac{1}{\sqrt{2J + 1}} \langle \psi_{JM} | [A^\dagger (ab; J_{ab}M_{ab}) \otimes \tilde{A}(cd; J_{cd}M_{cd})]^J | \psi_{JM} \rangle,$$

(A.4)

where $A^\dagger$ and $A$ are the same as in Eqs. (5, 6). For the factorization in Eq. (13), the $\text{TBTD}$ for a two-body interaction $V_{ij}(abcd)$ is defined as follows

$$\text{TBTD} \equiv \sqrt{\frac{2J_{ab} + 1}{2J_f + 1}} \text{TBTD}(f; abcd; J_{ab}J_{cd}; 0),$$

(A.5)

where only the term with $\lambda = 0, J_{cd} = J_{ab}, M_{cd} = M_{ab}, J_f = J_i, M_f = M_i$ is needed for Eq. (13) due to the symmetry.

Appendix B. Sample codes

We provide the following sample codes in the GitHub repository [55] for reproducibility of the results discussed in the main text:

(a) 10 lowest states of a target nucleus
(b) 10 lowest states of the target with a specific $J$
(c) EC estimates of 10 lowest states of the target with the $J$
(d) (b) with the preprocessing

A short guide for installation and running the sample program is given below. First, one should prepare Julia environment, which is easily done by downloading a binary from the official website of the Julia Language or $\texttt{wget}$ in a Linux-like environment. Second, make a clone of ShellModel.jl repository and execute $\texttt{julia src/package_install.jl}$ to install other open Julia packages. Then, execute e.g., $\texttt{julia -t 12 sample_run.jl}$, where “-t” specifies the number of threads to be used. Documentation for major functions is also available in the repository.

For the transition density matrices needed for (c), we provide the one with the smaller sample size $N_T = 50$, because the files for $N_T > 50$ are too large to share on the GitHub. The transition densities for a larger sample size are available upon request from the corresponding author.

Appendix C. MCMC settings

In this section, we summarize the detailed MCMC settings for the results in Sec. 3.3 and, the distributions of the posterior and the energy eigenvalues.

Under the likelihood (Eq. (14)) and the prior (Eq. (16)), we generated 100,000 MCMC samples after 10,000 burn-in. The sampling is done on the 66-dimensional parameter space in the isospin formalism, and it can be extended to the so-called proton-neutron formalism or some linear combination of two-body matrix elements. The variance for the proposal is fixed 6.2-2, and the resultant acceptance rate was around 0.23, which is close to the well-known asymptotic optimal value 0.234 for independent and identically distributed Gaussian [54]. When discussing highly multimodal distributions, it is recommended to use more efficient sampling schemes, such as parallel tempering and adaptive schemes to design the proposal distribution.

A full version of Fig. 3 the MCMC posterior of the 66 parameters, is shown in Fig. C.6. While most distributions are almost the same as the prior, some are slightly modified through the likelihood term. The resultant distribution of energy eigenvalues for the twelve data is summarized in Fig. C.7. The dot and cross symbols are experimental data and the exact value under the USDB interaction, respectively. The three independent runs are depicted as histograms with a bin width of 500 keV, but the lines in the plot are hardly distinguishable. This means that the MCMC results are converged within a Monte Carlo error.

Appendix D. Convergence of the Lanczos iterations

In Figs. D.8 D.10 the convergence patterns for 25Mg, 26Al, and 28Si are shown as a function of the number of Lanczos iterations. The lines and symbols are the same as Fig. 3 for 24Mg. As a whole, the block algorithm and the preprocessing reduce the number of iterations of the Lanczos method, which is the most time-consuming part of the calculations. As mentioned in the main text, however, the preprocessing can deteriorate the convergence compared to the case of random initial vectors if the sampled wave functions are blind to the higher excited states.
Figure C.6: Probability density distributions, prior (dashed line) and posterior (histograms), for the 66-dimensional parameters. The lines, symbols, and histograms are the same as Fig. 3. The first three out of the 66 parameters are SPEs of $1/s_1^2$, $0/d_3^2$, $0/d_5^2$, respectively. The correspondence for the remaining TBMEs ($n = 4 – 66$) can be found by the file `usdb.int` in the repository [35].

Figure C.7: Distributions of energy eigenvalues of twelve data of $^{28}$Si. The values are in units of MeV. All the histograms are normalized and drawn by the 100,000 MCMC samples with a bin width of 500 keV. The dot and cross symbols are experimental data and the exact value under the USDB interaction, respectively. To check the convergence, three independent MCMC runs are plotted.
Figure D.8: Convergence pattern of the block Lanczos iterations for $J = 1/2$ states of $^{25}\text{Mg}$. The lines and symbols are the same as Fig. D.5.

Figure D.9: Convergence pattern of the block Lanczos iterations for $J = 0$ states of $^{26}\text{Al}$. The lines and symbols are the same as Fig. D.5.

Figure D.10: Convergence pattern of the block Lanczos iterations for $J = 0$ states of $^{28}\text{Si}$. The lines and symbols are the same as Fig. D.5.

References

[1] B. A. Brown, The nuclear shell model towards the drip lines, Progress in Particle and Nuclear Physics 47 (2) (2001) 517 – 599. doi:https://doi.org/10.1016/S0146-6410(01)00159-4 URL http://www.sciencedirect.com/science/article/pii/S0146641001001594

[2] E. Caurier, G. Martinez-Pinedo, F. Nowacki, A. Poves, A. P. Zuker, The shell model as a unified view of nuclear structure, Rev. Mod. Phys. 77 (2005) 427–488. doi:10.1103/RevModPhys.77.427 URL https://link.aps.org/doi/10.1103/RevModPhys.77.427

[3] T. Otsuka, A. Gade, O. Sorlin, T. Suzuki, Y. Utsuno, Evolution of shell structure in exotic nuclei, Rev. Mod. Phys. 92 (2020) 015002. doi:10.1103/RevModPhys.92.015002 URL https://link.aps.org/doi/10.1103/RevModPhys.92.015002

[4] E. Epelbaum, H.-W. Hammer, U.-G. Meißner, Modern theory of nuclear forces, Rev. Mod. Phys. 81 (2009) 1773–1825. doi:10.1103/RevModPhys.81.1773 URL http://link.aps.org/doi/10.1103/RevModPhys.81.1773

[5] R. Machleidt, D. Entem, Chiral effective field theory and nuclear forces, Phys. Rept. 503 (1) (2011) 1 – 75. doi:10.1016/j.physrep.2011.02.001 URL http://www.sciencedirect.com/science/article/pii/S0370157311000457

[6] S. K. Bogner, H. Herrgott, J. D. Holt, A. Schwenk, S. Binder, A. Calci, J. Langhammer, R. Roth, Nonperturbative shell-model interactions from the in-medium similarity renormalization group, Phys. Rev. Lett. 113 (2014) 142501. doi:10.1103/PhysRevLett.113.142501 URL https://link.aps.org/doi/10.1103/PhysRevLett.113.142501

[7] S. R. Stroberg, H. Herrgott, J. D. Holt, S. K. Bogner, A. Schwenk, Ground and excited states of doubly open-shell nuclei from ab initio valence-space hamiltonians, Phys. Rev. C 93 (2016) 051301.
[40] M. Honma, T. Otsuka, T. Mizusaki, M. Hjorth-Jensen, New effective interaction for \( ^{16}\text{O} \) shell nuclei, Phys. Rev. C 80 (2009) 064323. doi:10.1103/PhysRevC.80.064323 URL https://link.aps.org/doi/10.1103/PhysRevC.80.064323

[41] A. Magilligan, B. A. Brown, New isospin-breaking “usd” hamiltonians for the sd shell, Phys. Rev. C 101 (2020) 064312. doi:10.1103/PhysRevC.101.064312 URL https://link.aps.org/doi/10.1103/PhysRevC.101.064312

[42] T. Mizusaki, K. Kaneko, M. Honma, T. Sakurai, Filter diagonalization of shell-model calculations, Phys. Rev. C 82 (2010) 024310. doi:10.1103/PhysRevC.82.024310 URL https://link.aps.org/doi/10.1103/PhysRevC.82.024310

[43] A. Ekström, G. Hagen, Global sensitivity analysis of bulk properties of an atomic nucleus, Phys. Rev. Lett. 123 (2019) 252501. doi:10.1103/PhysRevLett.123.252501 URL https://link.aps.org/doi/10.1103/PhysRevLett.123.252501

[44] R. Furnstahl, A. Garcia, P. Millican, X. Zhang, Efficient emulators for scattering using eigenvector continuation, Physics Letters B 809 (2020) 135719. doi:https://doi.org/10.1016/j.physletb.2020.135719 URL https://www.sciencedirect.com/science/article/pii/0370269320305220

[45] S. König, A. Ekström, K. Hebeler, D. Lee, A. Schwenk, Eigenvector continuation as an efficient and accurate emulator for uncertainty quantification, Physics Letters B 810 (2020) 135814. doi:https://doi.org/10.1016/j.physletb.2020.135814 URL https://www.sciencedirect.com/science/article/pii/0370269320306171

[46] D. Bai, Z. Ren, Generalizing the calculable r-matrix theory and eigenvector continuation to the incoming-wave boundary condition, Phys. Rev. C 103 (2021) 014612. doi:10.1103/PhysRevC.103.014612 URL https://link.aps.org/doi/10.1103/PhysRevC.103.014612

[47] S. Wesołowski, I. Svensson, A. Ekström, C. Forsén, R. J. Furnstahl, J. A. Melendez, D. R. Phillips, Fast & rigorous constraints on chiral three-nucleon forces from few-body observables (2021). [arXiv:2104.04441]

[48] P. Demol, T. Duguet, A. Ekström, M. Frosini, K. Hebeler, S. König, D. Lee, A. Schwenk, V. Somà, A. Tichai, Improved many-body expansions from eigenvector continuation, Phys. Rev. C 101 (2020) 041302. doi:10.1103/PhysRevC.101.041302 URL https://link.aps.org/doi/10.1103/PhysRevC.101.041302

[49] P. Demol, M. Frosini, A. Tichai, V. Somà, T. Duguet, Bogoliubov many-body perturbation theory under constraint, Annals of Physics 424 (2021) 168358. doi:10.1016/j.aop.2020.168358 URL https://www.sciencedirect.com/science/article/pii/000349162030292X

[50] A. Sarkar, D. Lee, Convergence of eigenvector continuation, Phys. Rev. Lett. 126 (2021) 032501. doi:10.1103/PhysRevLett.126.032501 URL https://link.aps.org/doi/10.1103/PhysRevLett.126.032501

[51] M. Hjorth-Jensen, T. T. Kuo, E. Osnes, Realistic effective interactions for nuclear systems, Physics Reports 261 (3) (1995) 125–270. doi:https://doi.org/10.1016/0370-1573(95)00012-6 URL https://www.sciencedirect.com/science/article/pii/0370157395000126

[52] A. De-Shalit, I. Talmi, Nuclear shell theory, Vol. 14, Academic Press, 2013.

[53] B. A. Brown, Lecture notes in nuclear structure physics, National Super Conducting Cyclotron Laboratory (2005).

[54] G. O. Roberts, A. Gelman, W. R. Gilks, Weak convergence and optimal scaling of random walk metropolis algorithms, The Annals of Applied Probability 7 (1)(1997) 110–120. URL http://www.jstor.org/stable/2245134