No Core CI calculations for light nuclei with chiral 2- and 3-body forces

Pieter Maris[1], **H Metin Aktulga**[2], **Sven Binder**[3], **Angelo Calci**[3], **Ümit V Çatalyürek**[4,5], **Joachim Langhammer**[3], **Esmond Ng**[2], **Erik Saule**[4], **Robert Roth**[3], **James P Vary**[1] **and Chao Yang**[2]

[1] Department of Physics and Astronomy, Iowa State University, Ames, IA 50011, USA
[2] Computational Research Division, Lawrence Berkeley National Lab, Berkeley, CA 94720, USA
[3] Institut für Kernphysik, Technische Universität Darmstadt, 64289 Darmstadt, Germany
[4] Dept. of Biomedical Informatics, The Ohio State University, Columbus, OH 43210, USA
[5] Dept. of Electrical and Computer Engineering, The Ohio State University, Columbus, OH 43210, USA

E-mail: pmaris@iastate.edu

Abstract. The atomic nucleus is a self-bound system of strongly interacting nucleons. In No-Core Configuration Interaction calculations, the nuclear wavefunction is expanded in Slater determinants of single-nucleon wavefunctions (Configurations), and the many-body Schrödinger equation becomes a large sparse matrix problem. The challenge is to reach numerical convergence to within quantified numerical uncertainties for physical observables using finite truncations of the infinite-dimensional basis space. We discuss strategies for constructing and solving the resulting large sparse matrices for a set of low-lying eigenvalues and eigenvectors on current multicore computer architectures. Several of these strategies have been implemented in the code MFDn, a hybrid MPI/OpenMP Fortran code for ab initio nuclear structure calculations that scales well to over 200,000 cores. We discuss how the similarity renormalization group can be used to improve the numerical convergence. We present results for excitation energies and other selected observables for ^{8}Be and ^{12}C using realistic 2- and 3-body forces obtained from chiral perturbation theory. Finally, we demonstrate that collective phenomena such as rotational band structures can emerge from these microscopic calculations.

1. No-Core Configuration Interaction approach

Solving for nuclear properties with realistic potentials, using a quantum many-particle framework that respects all the known symmetries of these potentials, is recognized to be computationally hard. A nucleus with Z protons and N neutrons is a self-bound quantum many-body system with $A = N + Z$ strongly interacting nucleons. The interactions are strong (otherwise nuclei would not be bound) and feature both attractive and repulsive contributions along with significant spin and angular momentum dependence. Furthermore there are both short-range and long-range terms in the interaction, and in addition to two-body forces (2NF) [1, 2, 3], one also needs suitable three-nucleon forces (3NF) [4, 5, 6, 7, 8] (and possibly higher A-body forces).

A commonly used approach to address this problem in nuclear physics is the Configuration Interaction (CI) method for solving the many-body nuclear Hamiltonian in a (sufficiently large)

Published under licence by IOP Publishing Ltd

single-particle basis space. In this approach, the many-body Schrödinger equation

$$H \ \Psi_i(\vec{r}_1, \ldots, \vec{r}_A) \ = \ E_i \ \Psi_i(\vec{r}_1, \ldots, \vec{r}_A) \qquad (1)$$

becomes a large sparse matrix problem with eigenvalues E_i. In No-Core CI (NCCI) calculations [9], the wavefunction Ψ of a nucleus consisting of A nucleons is expanded in an A-body basis of Slater determinants Φ_k of single-particle wavefunctions ϕ_i

$$\Psi(\vec{r}_1, \ldots, \vec{r}_A) = \sum c_k \Phi_k(\vec{r}_1, \ldots, \vec{r}_A) \,, \qquad (2)$$

with $\Phi_k(\vec{r}_1, \ldots, \vec{r}_A) = \mathcal{A}[\phi_{n_1 l_1 j_1 m_1 tz_1}(\vec{r}_1)\, \phi_{n_2 l_2 j_2 m_2 tz_2}(\vec{r}_2) \ldots \phi_{n_A l_A j_A m_A tz_A}(\vec{r}_A)]$. Here $\mathcal{A}$ is the antisymmetrization operation that insures compliance with the Pauli exclusion principle for identical fermions. Conventionally, one uses a harmonic oscillator (HO) basis for the single-particle wavefunctions, but it is straightforward to extend this approach to a more general single-particle basis [10]. The single-particle wavefunctions are labelled by the quantum numbers n, l, j, m, and tz, where n and l are the radial and orbital HO quantum numbers, with $N = 2n+l$ the number of HO quanta; j is the total single-particle spin, with m its projection along the z-axis; and tz is the isospin projection ($+\frac{1}{2}$ for protons and $-\frac{1}{2}$ for neutrons). The many-body basis states have well-defined total spin projection M, which is simply the sum of m_k of the single-particle states, $M = \sum m_k$, hence the name M-scheme. However, the many-body basis states do not have a well-defined total spin J. Some of the benefits of this scheme are that it is very simple to implement, and that in two runs (one for positive and one for negative parity states), we get the complete low-lying spectrum, including the ground state, even if the spin of the ground state is unknown.

In a complete basis, this method would give exact results for a given input interaction V. However, practical calculations can only be done in a finite-dimensional truncation of a complete basis. Different truncation schemes have different convergence rates: Full Configuration Interaction employs a widely-know truncation in which all many-body basis states that can be constructed from a finite set of single-particle states are retained. However, this truncation method has a rather slow convergence rate in typical ab initio nuclear structure calculations [11, 12]. The so-called $N_{\max}$ truncation, in which the total number of HO quanta in a basis state is limited: $\sum N_k \leq N_0 + N_{\max}$, is generally much more efficient for NCCI calculations. Here, N_k is the number of quanta of each single-particle state in the many-body basis state; N_0 is the minimal number of quanta for that nucleus; and $N_{\max}$ is the truncation parameter. For HO single-particle states, or more general, for any basis in which the single-particle states have radial and orbital quantum numbers n and l, we have $N_k = 2n_k + l_k$. Furthermore, in combination with a HO basis as single-particle basis, this $N_{\max}$ truncation leads to an exact factorization of the center-of-mass wavefunction and the relative wavefunction.

2. Computational Challenges and Solutions

For a very light nucleus like ^{4}He or ^{6}Li one can achieve convergence of the ground state energy and other observables with suitable 2NFs by simply going to a sufficiently large basis, but this is not practical for larger nuclei and with 3NFs. The computational effort depends not so much on the dimension of the matrix, but rather on the actual number of nonzero matrix elements in the sparse matrix. The number of nonzero matrix elements in a calculation including 3NFs is between one and two orders of magnitude larger than the number of nonzero matrix elements in a calculation for the same nucleus in the same basis space, but using 2NFs only [13].

For p-shell nuclei ($N, Z \leq 8$), the dimension of the M-scheme basis space at $N_{\max} = 8$ reaches about one billion basis states. With 2NFs plus 3NFs, the resulting symmetric matrix contains up to 40 trillion nonzero real matrix elements in the upper (or equivalently, in the lower) triangle; see figure 1. Storing these nonzero matrix elements in compressed-row format (or

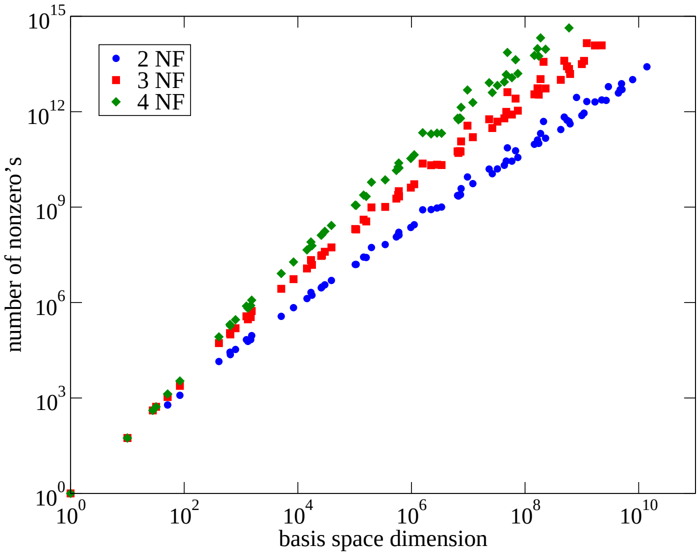

Figure 1. Number of nonzero matrix elements as function of the basis dimension, with 2-body forces (2NF, blue circles,), 2- and 3-body forces (3NF, red squares) and up to 4-body forces (4NF, green diamonds) in an M-scheme basis for $N = Z$ nuclei with $M = 0$.

equivalently, compressed-column format) using single-precision requires an aggregate memory of about 320 TB. That is slightly over half the total memory of Jaguar, a Cray XK6 with 18,688 16-core AMD processors at Oak Ridge National Laboratory (currently being upgraded to Titan, a Cray XK7); slightly less than half the total memory of Mira, an IBM Blue Gene/Q with 49,152 16-core processors at Argonne National Laboratory; and about a quarter of the total memory of the K computer, a Fujitsu supercomputer with 88,128 SPARC64 processors at the RIKEN Advanced Institute for Computational Science in Kobe, Japan. In order to make use of such resources, which are available for open science, we clearly need a suitable code that distributes the matrix evenly over the available processors, and that performs well at such large scales.

For the results presented here we used the code MFDn, which is a hybrid MPI/OpenMP parallel Fortran 90 code for nuclear structure calculations, that has been in development for over two decades. Significant improvements in its performance have been made over the last five years [14, 15, 16, 17] under the DOE SciDAC-2 UNEDF project. MFDn constructs the many-body basis states in M-scheme and the corresponding many-body matrix, and solves for the lowest eigenstates using an iterative Lanczos algorithm. Each Lanczos iteration consists of a sparse matrix-vector multiplication (SpMV) followed by orthogonalization. At the end of a run MFDn writes the eigenvectors to file, and evaluates selected physical observables which can be compared to experimental data (the wavefunctions themselves are not experimentally observable). It also writes out the one-body density matrix elements (in the underlying single-particle basis), which can be used for further analysis. By far the most compute-intensive parts of the calculation are constructing the large sparse matrix and obtaining the lowest eigenvalues and eigenvectors of this matrix using a Lanczos algorithm.

2.1. Two-dimensional distribution of sparse matrix

Each vector is distributed over n processors, and the sparse symmetric matrix itself is distributed in a two-dimensional fashion over $n(n + 1)/2$ processors. We keep the matrix in core, using compressed-row or compressed-column format, and we store only half the matrix, because the matrix is symmetric; we use the same data structure for both a SpMV and a transpose SpMV. The local work load of the SpMV is proportional to the number of nonzero matrix elements on each processor. Thus we want to distribute the matrix in such a way that each processor has approximately the same number of nonzero matrix elements. Note that the overall performance also depends on the communication overhead, and that the mapping of the matrix blocks onto

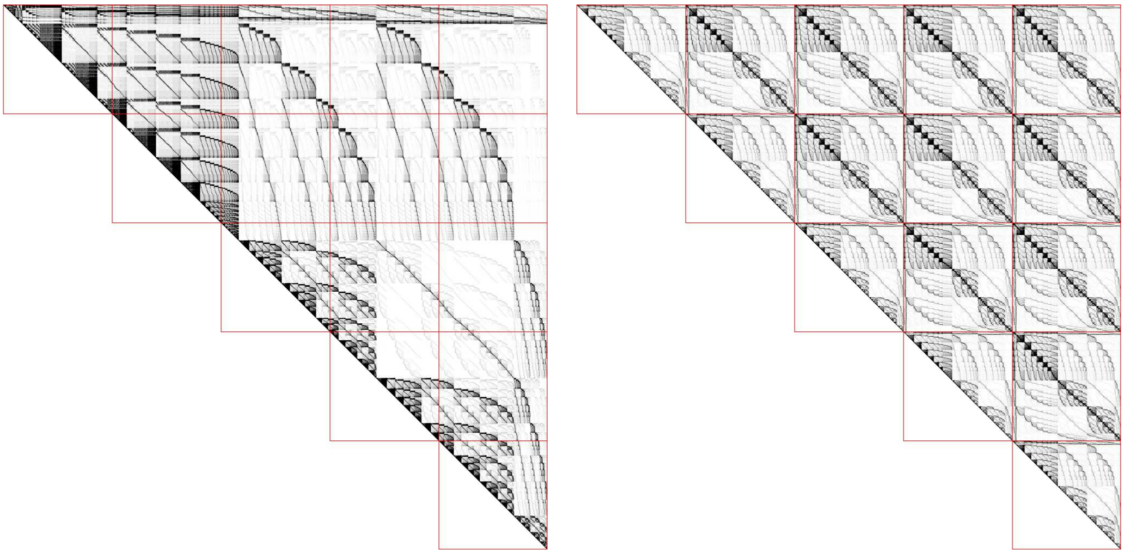

Figure 2. Sparsity structure for ^{8}Be at $N_{\rm max} = 4$ with 3NF, with a dimension of 143,792 M-scheme basis states and 402,513,272 nonzero matrix elements in the upper triangle: sparsity structure on a single processor (left); and evenly distributed over 15 processors after a round-robin distribution of the basis states over 5 processor (right).

the MPI processors has a large influence on this communication overhead [16, 17].

On a single processor, a lexicographical enumeration of the M-scheme many-body basis states leads to a very structured sparse matrix, as can be seen in the left panel of figure 2, although it is a rather nontrivial structure. A simple distribution of this structured matrix over 15 processors (see the red grid in figure 2) leads to rather poor load balancing and therefore very poor scaling.

On the other hand, with a round-robin distribution of individual M-scheme basis states over n processors, we achieve almost perfect load balancing over $n(n+1)/2$ processors for the matrix, both in memory usage and the number of local floating-point operations during the Lanczos iterations, as is indicated in the right panel of figure 2. This scheme has been implemented and used in various versions of MFDn, up through the current production Version 13, and indeed leads to very good scaling of the SpMV – in fact, sometimes better than ideal scaling, as we will show below. However, a disadvantage of a round-robin distribution of individual basis states is that on any given processor there is little correlation between neighboring states. Although this distribution leads to an overall sparsity pattern for each submatrix on one of the $n(n+1)/2$ processors that looks similar to that of the entire matrix on a single processor, see figure 2, at the level of individual matrix elements we are effectively dealing with a randomly sparse matrix; see the top panel of figure 3. For large runs with 3NFs, we have sparse matrices with less than 1 in 10^3 nonzero matrix elements. Determining exactly which matrix elements are nonzero becomes an extremely time-consuming task for such large sparse matrices [14]. Furthermore, any local SpMV with a randomly sparse matrix will suffer from poor cache performance, in particular if the input and output vectors are too large to remain in cache.

In the latest version of MFDn (Version 14, currently under development), we distribute groups of M-scheme basis states in a round-robin fashion over the n processors, instead of individual basis states. One of the advantages of this scheme is that on any given processor at least some of the neighboring basis states are correlated (namely those within a group), leading to a more regular and predictable sparsity structure down to the level of individual matrix elements; see the bottom panel of figure 3. This greatly improves the efficiency of

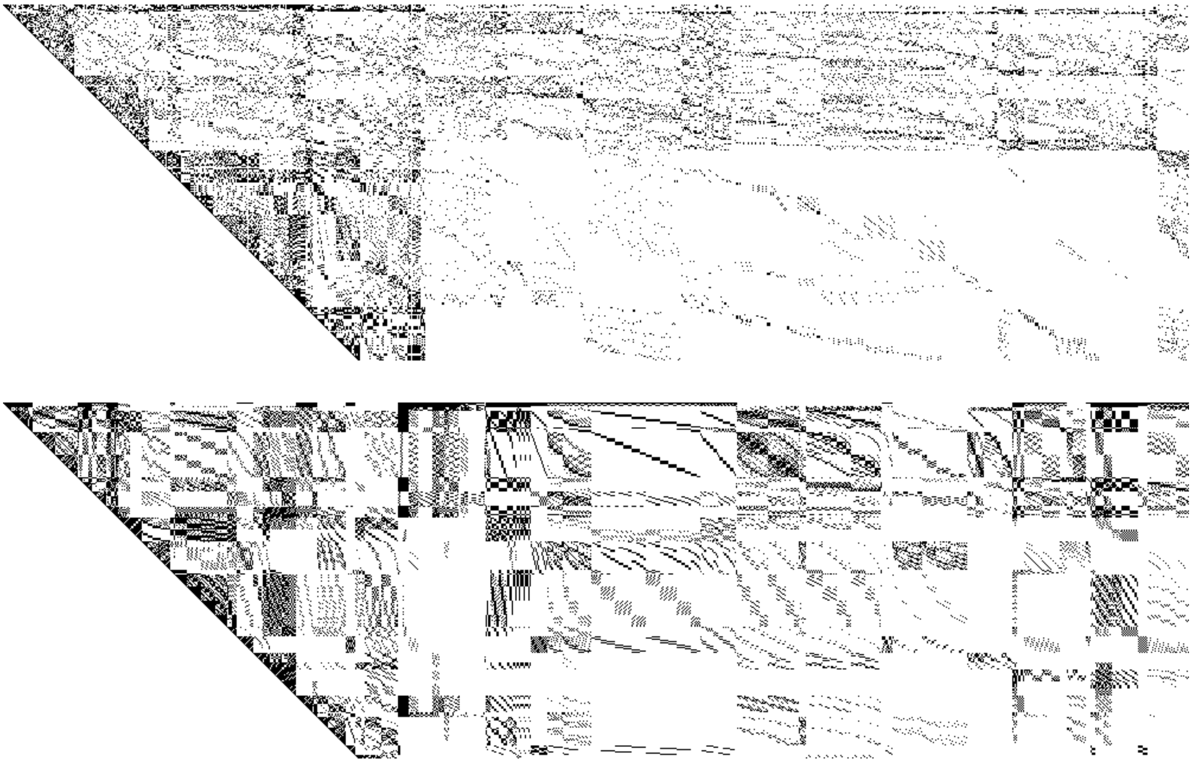

Figure 3. Sparsity structure for ^{8}Be at $N_{\max} = 4$ with 3NF on 120 processors: detailed structure of the upper-left (1000×300) submatrix on the first (diagonal) processor (each dot represents a single matrix element). Top: using round-robin distribution of individual M-scheme basis states (Version 13 of MFDn); Bottom: using round-robin distribution of groups of states (Version 14, under development).

determining which matrix elements are nonzero, while at the same time improving the cache performance. The main disadvantage is that the load-balancing is not as good as with the round-robin distribution of individual states. By changing the definition of a 'group of states' one can trade-off between more structure in the matrix but less load-balanced, and vice versa: smaller groups lead to better load-balancing but less structure, larger groups give less load-balancing but more structure. The 'ideal' definition of a group of states depends on a combination of the nucleus under consideration, the truncation and basis space, and the specific hardware for the calculation (available aggregate memory, memory per core, cache sizes, and memory bandwidth).

2.2. Input 3NF matrix elements

Another recent improvement concerns the input 3NF matrix elements. For the construction of the many-body matrix we need 2NF and 3NF matrix elements in M-scheme in a proton-neutron basis. These can be read in from file either in M-scheme or in a more compact representation known as the coupled-J (or coupled-JT) format [18]. The angular momenta j of the 2 (or 3) interacting nucleons are coupled to form 2-nucleon (or 3-nucleon) states of well-defined combined angular momentum J, using Clebsch–Gordan coefficients $(j_\alpha m_\alpha, j_\beta m_\beta | JM)$

$$|n_\alpha l_\alpha j_\alpha m_\alpha; n_\beta l_\beta j_\beta m_\beta\rangle \;=\; \sum_J (j_\alpha m_\alpha, j_\beta m_\beta | JM) \; \left|(n_\alpha l_\alpha j_\alpha, n_\beta l_\beta j_\beta)^J\right\rangle \,, \qquad (3)$$

with $M = m_\alpha + m_\beta$, and the sum runs over the allowed values $|j_\alpha - j_\beta| \le J \le j_\alpha + j_\beta$. Since the interactions are rotationally invariant, and thus preserve total angular momentum J, we only have nonzero matrix elements between 2-nucleon (or 3-nucleon) states of the same combined angular momentum J. (In addition to combining the angular momenta j of the interacting nucleons to a total angular momentum J, we can also combine the isospin $t = \frac{1}{2}$ of each nucleon to a total isospin of $T = 0$ or 1 in the 2NF case, or to $T = \frac{1}{2}$ or $\frac{3}{2}$ in the 3NF case.)

Thus for 2NFs we can either read in matrix elements between two nucleons in M-scheme

$$\langle n_\alpha l_\alpha j_\alpha m_\alpha ; n_\beta l_\beta j_\beta m_\beta | \, V \, | n_\gamma l_\gamma j_\gamma m_\gamma ; n_\delta l_\delta j_\delta m_\delta \rangle \ ,$$

with $m_\alpha + m_\beta = m_\gamma + m_\delta$, or we can read in matrix elements using the in coupled-J scheme

$$\left\langle (n_\alpha l_\alpha j_\alpha , n_\beta l_\beta j_\beta)^J \right| V \left| (n_\gamma l_\gamma j_\gamma , n_\delta l_\delta j_\delta)^J \right\rangle \ .$$

The latter is a more compact form and requires less memory, but when we construct the many-body matrix we need to decouple (or 'decompress') the coupled-J matrix elements. That is, for every nonzero matrix element needed for the construction of the many-body matrix, we need to perform a sum over the allowed values of J

$$\begin{aligned} \langle n_\alpha l_\alpha j_\alpha m_\alpha ; n_\beta l_\beta j_\beta m_\beta | V | n_\gamma l_\gamma j_\gamma m_\gamma ; n_\delta l_\delta j_\delta m_\delta \rangle &= \\ \sum_J (j_\alpha m_\alpha , j_\beta m_\beta | JM) \ (j_\gamma m_\gamma , j_\delta m_\delta | JM) \ & \left\langle (n_\alpha l_\alpha j_\alpha , n_\beta l_\beta j_\beta)^J \right| V \left| (n_\gamma l_\gamma j_\gamma , n_\delta l_\delta j_\delta)^J \right\rangle \ , \end{aligned} \tag{4}$$

with $|j_\alpha - j_\beta| \leq J \leq j_\alpha + j_\beta$, which involves either (repeatedly) calculating the corresponding Clebsch–Gordan coefficients, or (more efficient in terms of number of floating-point operations) use a look-up table for these Clebsch–Gordan coefficients.

For 3NFs we end up with triple sums over allowed J values. Schematically

$$\langle j_\alpha j_\beta j_\gamma | V | j_{\alpha'} j_{\beta'} j_{\gamma'} \rangle \ = \ \sum_{J_\delta , J_{\delta'} , J} \cdots \left\langle \left((j_\alpha , j_\beta)^{J_\delta} , j_\gamma \right)^J \right| V \left| \left((j_{\alpha'} , j_{\beta'})^{J_{\delta'}} , j_{\gamma'} \right)^J \right\rangle \tag{5}$$

with the dots representing a product of four Clebsch–Gordan coefficients. In principle, one could extend this further: For 4NFs one would get a quintuple sum involving a product of six Clebsch–Gordan coefficients.

Not surprisingly, the number of input matrix elements grows with the basis space. For p-shell nuclei ($N, Z \leq 8$), the number of 3NF input matrix elements in M-scheme in a proton-neutron basis is 741,823,056 at $N_{\max} = 6$, corresponding to an input file of 3 GB (using single-precision real numbers), and at $N_{\max} = 8$ the input file is 33 GB, containing over 8 billion input M-scheme matrix elements. Thus, up to $N_{\max} = 6$ we can work with the input data in M-scheme on current architectures, but at $N_{\max} = 8$ and above we quickly run out of memory. We can deal with such large input files by processing, for example, the input data for an $N_{\max} = 8$ run in eight to twelve 'chunks' of a few GB each, but this is not very efficient, as we demonstrate in the next section, and becomes prohibitively expensive above $N_{\max} = 8$.

In contrast, the number of 3NF input matrix elements in coupled-JT format is only 111,804,075 at $N_{\max} = 8$ for p-shell nuclei, that is, less than 0.5 GB. Even at $N_{\max} = 10$, the total memory required for the 3NF input in coupled-JT format is only about 2.4 GB [18], which is manageable on current multi-core (and future many-core) architectures. The price we pay for using the coupled-JT format is that we need to decompress the input data whenever we need to evaluate a matrix element, but this price may be tolerable since the trend in hardware is that threads for floating-point operations are becoming more abundant, whereas memory is expected to remain at a premium.

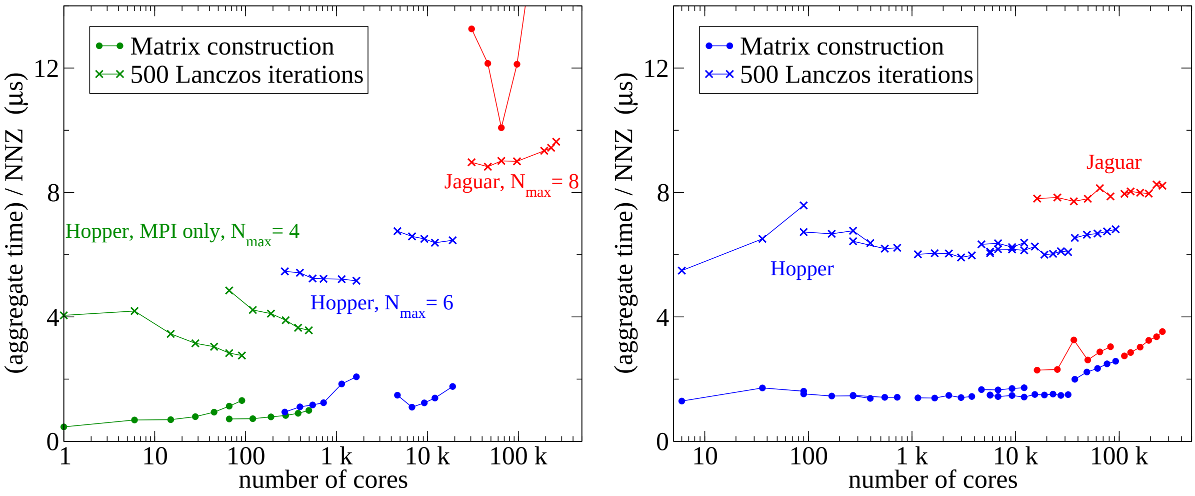

Figure 4. Scaling of MFDn Version 13 (left) and Version 14 (right) on Hopper at NERSC with MPI only (green symbols) and hybrid MPI/OpenMP (blue symbols), and on Jaguar at ORNL (red symbols) with two- plus three-body forces. (For technical reasons, we use the PGI compiler for Version 13 and the gnu compiler with Version 14, both with optimization.)

2.3. Strong and weak scaling

In figure 4 we show the strong and weak behavior of MFDn on Hopper at the National Energy Research Scientific Computing Center (NERSC) and on Jaguar at Oak Ridge National Laboratory (ORNL). Since the cost for the SpMV is largely proportional to the number of nonzero matrix elements, we show the aggregate time to construct the matrix (defined as wall clock time multiplied by the number of cores) divided by the number of nonzero matrix elements. Also the cost for the construction of the matrix is proportional to the number of nonzero matrix elements, once the sparsity structure is known (which is not included in the timing data of figure 4). Ideal scaling corresponds to a straight horizontal line in these plots; and the lower this line is, the better the performance. Symbols connected by a line are for the same nucleus and basis space, and thus indicate the strong scaling behavior.

Hopper is a Cray XE6 at NERSC consisting of 6,384 nodes with two 12-core AMD 'MagnyCours' processors per node and 32 GB memory per node. Each of the 24 cores have access to all of the 32 GB memory on the node, but the memory access time is not uniform: the compute nodes are so-called NUMA nodes (Non Uniform Memory Access). Each node has four memory banks of 8 GB each; 6 cores share a memory bank (as well as the L3 cache). Accessing memory from one of the three other memory banks is significantly slower than accessing local memory. For MFDn we therefore typically use four MPI processors per node, with six threads per MPI processor, on Hopper, though for relatively small runs our MPI-only implementation performs better. Jaguar was a Cray XK6 at ORNL, and consisted of 18,688 nodes with a 16-core AMD 'Interlagos' processor and 32 GB of memory per node. Again, each node is a NUMA node, with, on Jaguar, two NUMA domains and 8 cores per NUMA domain. On Jaguar we therefore typically ran with two MPI processors per node and 8 threads per MPI processor. Both Hopper and Jaguar use the Gemini interconnect between nodes. (Jaguar is currently being upgraded to Titan by adding a GPU to each node.)

In the left panel of figure 4 we see that for relatively small $N_{\max} = 4$ runs on Hopper using MPI only, the Lanczos iteration time actually decreases with increasing number of processors (see the green connected crosses): we have better than ideal strong scaling for these cases! It is not entirely clear what causes this behavior, but most likely it is due to a better cache performance as the local matrix blocks, as well as the local vector segments, on each processor get smaller. The minimal number of processors needed for a specific calculation is to a large extent dictated by the memory needed to store the matrix (distributed evenly over all processors). And since we perform both a SpMV and a transpose SpMV with the same compressed-row data structure for the matrix, our cache performance is far from ideal on the minimal number of processors. As we increase the number of processors for the same problem (strong scaling), the size of the matrix blocks on each processor decreases proportionally, and more of the in- and out-vectors of the SpMV can be kept in cache. Hence the cache performance during the Lanczos iterations is likely to improve as the number of processors increases for a fixed problem size.

For larger problem sizes using a hybrid MPI/OpenMP implementation (red and blue symbols), we do not see this effect. Most likely, the in- and out-vectors of the SpMV remain too large to fit in cache, even as we increase the number of processors: if we distribute the matrix such that we have four matrix blocks (one per NUMA node) on each node on Hopper, instead of 24 matrix blocks (one per core), then the dimensions of those blocks, and thus the length of the local vector segments, will be significantly larger. In addition, the memory needed for the local vector segments increases relative to the memory needed to store the local matrix block as both the problem size and the number of processors increase.

The scaling of the construction of the matrix (solid circles) is not as good as the scaling during the Lanczos iterations. This is likely to be caused by the fact that in Version 13 (left panel of figure 4) the time to distribute the input M-scheme matrix elements from the n processors on which they are read in to all $n(n+1)/2$ processors is included in the time to construct the matrix (the actual IO time is not included in figure 4). For $N_{\max} = 6$ calculations, with an M-scheme input file size of 3 GB, this is not too much of an issue, but for $N_{\max} = 8$ calculations, with an M-scheme input file size of 33 GB, we see that the matrix construction suddenly requires an order of magnitude more time than at $N_{\max} = 4$ and $N_{\max} = 6$ (red solid circles). This is partly due to the amount of input data to be redistributed (33 GB), and partly because we have to go through the construction of the matrix typically 8 to 12 times in order to process all input data.

The 3NF input matrix elements in coupled-JT format, as described in the previous subsection, resolves this issue of the size of the input file. Indeed, with Version 14 (right panel of figure 4) the construction of the matrix scales almost perfectly from 6 to 30,000 cores. The aggregate time is about twice as large as the most efficient $N_{\max} = 4$ and $N_{\max} = 6$ construction times using the M-scheme input matrix elements. Given the fact that we have to decouple the input matrix elements for every nonzero matrix element in the many-body matrix, see (5), this is actually a rather modest increase in aggregate time.

The scaling of the construction of the matrix breaks down above 40,000 cores, both on Hopper and Jaguar. Since there is no communication in this phase (at least not in Version 14 of MFDn), this has to be due to a load-imbalance. There are two sources of load imbalance: one is simply the number of nonzero matrix elements, and the other is the fact that for a subset of the nonzero matrix elements, we have to perform a single, double, or even triple sum over 'spectator' nucleons. In particular, all diagonal matrix elements involve such a triple sum. We are currently investigating this load imbalance problem.

Note that the Lanczos iterations with Version 14 show excellent strong and weak scaling over the entire range from 6 to 300,000 cores. Below 1,000 cores, the MPI-only implementation of Version 13 is more efficient, but for large runs, above 10,000 cores, Version 14 performs better, both in the construction of the matrix and in the Lanczos iterations, than Version 13. In addition, the time needed to determine the sparsity structure of the sparse matrix is typically, for large runs, an order of magnitude smaller with Version 14 than with Version 13, due to fact that Version 14 works with groups of M-scheme basis states, rather than individual states.

3. Convergence and renormalization of nuclear interactions

Even with all the recent improvements in code performance and efficiency, combined with the rapid increase in available computing power, convergence in a finite HO basis remains a challenge. Realistic 2NF potentials, such as Argonne V18 [1] and chiral N^{3}LO [2, 3], generate a combination of short-range and long-range correlations that cannot be represented in many-nucleon HO bases accessible at present computers. In order to account for the strong short-range (high-momenta) correlations and to improve convergence with increasing basis spaces, one can utilize a renormalization procedure specified by a unitary transformation of the Hamiltonian to a renormalized Hamiltonian

$$H_{\text{renormalized}} \quad = \quad U^\dagger \; H \; U \, , \tag{6}$$

that softens the interactions and generates effective operators for observables, while preserving all experimental quantities in the low-energy domain. The derived 'effective' interactions still act among all A nucleons and preserve all the symmetries of the initial 2NF and 3NF interactions.

One of these renormalization procedures is based on the Similarity Renormalization Group (SRG) [19, 20]. In the SRG approach the unitary transformation of the Hamiltonian is formulated in terms of a flow equation

$$\frac{dH_\alpha}{d\alpha} \quad = \quad [\eta_\alpha, H_\alpha] \; , \tag{7}$$

with a continuous 'SRG flow parameter' α, starting from $\alpha = 0$ fm^4 for the initial Hamiltonian $H_{\alpha=0}$. The physics of the SRG evolution is governed by the choice of the generator η_α. With a suitable choice of η_α, one can consistently evolve 2NFs and 3NFs (and even higher-body forces) to soften the short-range repulsion and tensor components of available initial interactions, so convergence of nuclear structure calculations is greatly accelerated [21, 22, 23, 24, 25, 26, 27]. The SRG evolution produces an 'effective' Hamiltonian that is independent of the nucleus and basis space, and generally enables smooth extrapolation of results [22, 27, 28].

A convenient and commonly used form in nuclear physics is

$$\eta_\alpha \quad = \quad m_N^2 \; [T_{\text{rel}}, H_\alpha] \; , \tag{8}$$

where T_{rel} is the kinetic energy operator in relative coordinates, and m_N is the nucleon mass. Thus the SRG evolution equation becomes

$$\frac{dH_\alpha}{d\alpha} \quad = \quad m_N^2 \; \Big[[T_{\text{rel}}, H_\alpha], H_\alpha \Big] \, . \tag{9}$$

This choice for the generator drives the renormalized Hamiltonian H_α towards the diagonal in a basis of eigenstates of the relative kinetic energy.

The SRG evolution generates many-body forces beyond those present in the initial Hamiltonian. The transformation is unitary only if all the induced terms up to the A-body level are included for an A-body calculation. Only in that case are the eigenvalues of the transformed Hamiltonian H_α identical to those of the initial Hamiltonian $H_{\alpha=0}$, independent of the SRG parameter α. These induced many-nucleon forces typically come in a hierarchy of decreasing strength with increasing number of interacting nucleons (at least for very small values of α). In practice one truncates the induced many-body terms either at the 2NF or at the 3NF level. A dependence of the obtained spectrum on α can therefore be attributed to the induced many-body forces that have been ignored in the calculation.

Calculations with the SRG evolved 2NF interactions, omitting induced 3NFs (as well as higher many-body forces) indeed show a significant dependence of the binding energies on the SRG flow parameter [22]. Most of this dependence is removed once the induced 3NF is properly taken into account [24, 25, 26, 27], at least for light nuclei and as long as the transformation is not pushed too far. With explicit 3NFs in the initial Hamiltonian however, there may be a need for induced 4-body forces [26].

4. Emergence of rotational band structures from ab initio calculations

Rotational bands arise quite naturally in α-cluster models for nuclei like ^{8}Be and ^{12}C, but it is not at all obvious that such structure also emerges from ab initio calculations, which describe these nuclei as bound states of eight and twelve nucleons respectively. The rotational energy for states in a rotational band with an axial symmetry is given by [29]

$$E_{\text{rotational}}(J) \;=\; \frac{\hbar^2}{2\mathcal{I}}\,\left(J(J+1)-K^2\right)\,, \tag{10}$$

where $\mathcal{I}$ is the moment of inertia, K is the projection of J along the intrinsic symmetry axis, and J is the total angular momentum. Furthermore, the rotational model predicts for the quadrupole moments of members of a rotational band

$$Q(J) \;=\; \frac{3K^2 - J(J+1)}{(J+1)(2J+3)}\,Q_0\,, \tag{11}$$

and for the strength of the E2 transitions within a rotational band

$$B(E2; i \to f) \;=\; \frac{5}{16\pi}\,\left(e\,Q_0\right)^2\,\left(J_i, K; 2, 0|J_f, K\right)^2\,, \tag{12}$$

where Q_0 is the 'intrinsic' quadrupole moment of the rotational band (in a body-fixed, non-inertial reference frame).

For even-even nuclei with a $J^\pi = 0^+$ ground state, we anticipate that we might have a $K = 0$ ground state rotational band. Rotational bands starting from $J = 0$ are restricted to even values of J only. Thus we expect a $J^\pi = 2^+$ and a 4^+ excited state, with a ratio of excitation energies $E_x(4)/E_x(2) = 10/3 = 3.33$.

Recent calculations using a realistic but nonlocal 2NF potential indicate that one does indeed obtain rotational band structures from ab initio nuclear structure calculations [30]. For the calculations presented here, we use the chiral N^{3}LO 2NF potential of [2] in combination with the chiral N^{2}LO 3NF potential of [5, 31]. For the two low-energy constants in the N^{2}LO 3NF potential we use the values that have been determined from the triton binding energy and β-decay [32]. In order to improve numerical convergence in a finite HO basis, we use the SRG renormalization procedure described in the previous section, and monitor the dependence of our results on the SRG parameter α.

4.1. Beryllium-8

Experimentally the two lowest excited states in ^{8}Be are a 2^+ and a 4^+ state, at 3.03 MeV and 11.35 MeV respectively, that is, with a ratio of excitation energies $E_x(4)/E_x(2) = 3.75$, somewhat larger than one would expect based on a rotational model.

We indeed find that the lowest two excited states are a 2^+ and a 4^+ state, as expected. With the chiral N^{3}LO 2NF potential, using SRG evolution including the induced 3NF, the excitation energies for these states are 3.2 MeV and 11.1 MeV respectively; see figure 5. We estimate these results to be converged to within about 5%, and they are virtually independent of the SRG parameter α. The ratio of the excitation energies is better converged, at $E_x(4)/E_x(2) = 3.45$, about halfway between the experimental value and that of a rotational model.

The effect of the chiral N^{2}LO 3NF on these excitation energies is rather small: including the initial 3NF we find 3.3 MeV and 11.5 MeV for the excitation energies of the 2^+ and 4^+ states respectively. The convergence of the excitation energies with the initial 3NF is similar to that with induced 3NFs only, and the ratio $E_x(4)/E_x(2)$ is converged to within a fraction of a percent at $N_{\text{max}} = 8$. For this ratio of the excitation energies we now find 3.48, only one percent larger than without initial 3NFs.

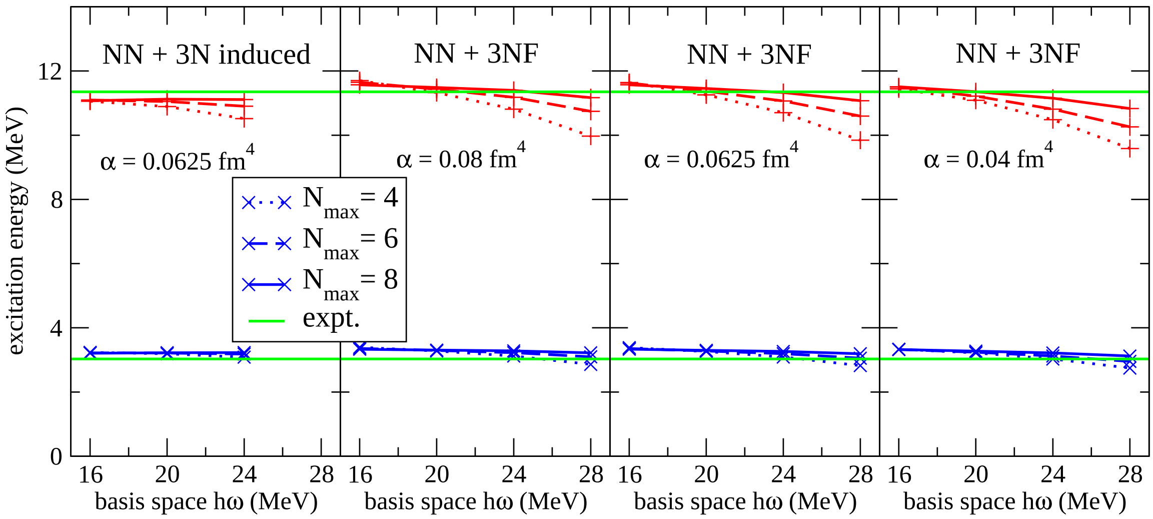

Figure 5. Excitation energies of the 2^+ (blue crosses) and 4^+ states (red plusses) for ^{8}Be with SRG evolved chiral N^{3}LO 2NF plus induced 3NF at $\alpha = 0.0625$ fm^4 (left-most panel) and with SRG evolved chiral N^{3}LO 2NF plus chiral N^{2}LO 3NF. Experimental values are indicated by the horizontal green lines.

The quadrupole moments and B(E2) transition strengths are not as well converged as the (excitation) energies. This is not surprising, since the E2 operator is a long-range operator, and therefore sensitive to the long-range behavior of the nuclear wavefunction. However, the long-range tail of the wavefunction is not well-represented in a HO basis: HO wavefunctions fall off like $\exp(-c\,r^2)$, whereas the nuclear wavefunction falls off like $\exp(-c\,r)$. Hence the convergence of observables like the quadrupole moments and B(E2)'s is rather slow in a HO basis.

Nevertheless, the ratios of the quadrupole moments seem to be in semi-quantitative agreement with the notion of these states being members of a rotational band. Although our results show a significant dependence on the basis space parameter $\hbar\omega$, see the middle panels of figure 6, the ratio $Q(4)/Q(2)$ is within about 10% of the rotational prediction of 1.27 at $N_{\rm max} = 8$. Note that the $\hbar\omega$ dependence is stronger with the initial chiral N^{2}LO 3NF than with the induced 3NFs only; and that the $\hbar\omega$ dependence is almost the same for the different values of α.

The B(E2) transition strength from the 2^+ to the ground state is in remarkable agreement with the rotational model, in terms of the quadrupole moment of the 2^+ state. Both without and with the initial chiral 3NFs the ratio B(E2)$/Q(2)^2$ is very well converged and almost independent of both $N_{\rm max}$ and $\hbar\omega$, and very close to the rotational value of 0.243; see the lower panels of figure 6. On the other hand, for the transition from the 4^+ to the 2^+ state the ratio B(E2)$/Q(4)^2$ is about 10% below what one would expect. Also note that this ratio appears to be somewhat better converged with initial chiral 3NFs than with the induced 3NFs only.

These results support the notion that the lowest 2^+ and the 4^+ excited states are, to a large extent, rotational excitations of the 0^+ ground state. Furthermore, the initial chiral 3NFs have little to no influence on the rotational nature of these excited states. Note that the total binding energy does depend on the 3NFs; also the question whether or not the ground state is bound with respect to breakup into two α particles with these interactions remains to be seen. At $N_{\rm max} = 8$ the binding energy of ^{8}Be is still below twice the binding energy of ^{4}He, calculated with the same interaction. Nevertheless, the excitation energies of lowest the 2^+ and the 4^+ states are very well converged, and in excellent agreement with the data.

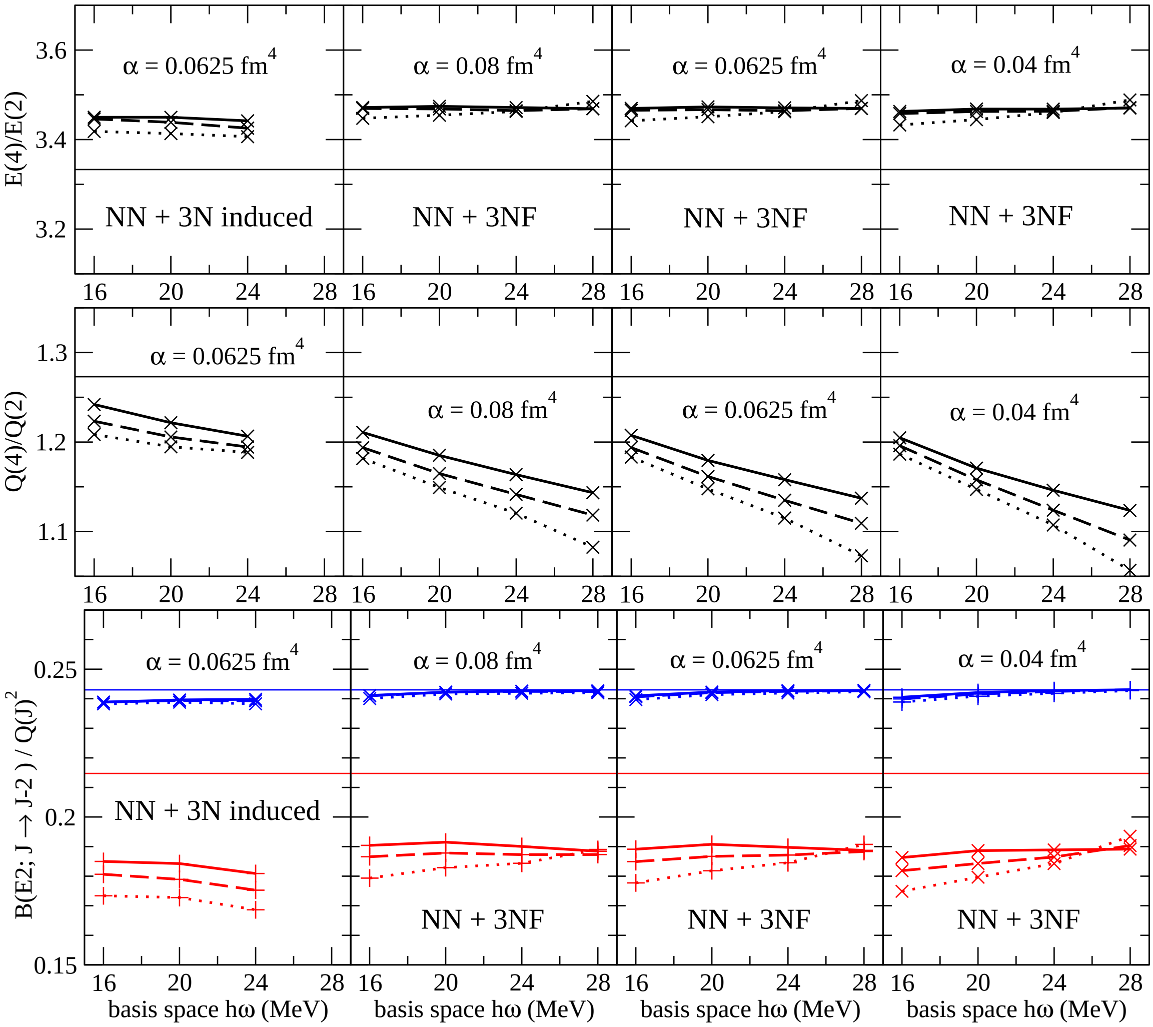

Figure 6. Ratios of results for ^{8}Be with SRG evolved chiral N^{3}LO 2NF plus induced 3NF at $\alpha = 0.0625$ fm^4 (left-most panel) and with SRG evolved chiral N^{3}LO 2NF plus chiral N^{2}LO 3NF. Horizontal lines represent the predictions from a rotational model. See figure 5 for the legend of the line styles. Top: excitation energies $E_x(4)/E_x(2)$; Middle: quadrupole moments $Q(4)/Q(2)$; Bottom: E2 transition strength B(E2)$/Q(J)^2$ for the $2^+ \to 0^+$ transition (blue crosses) and for the $4^+ \to 2^+$ transition (red plusses).

4.2. Carbon-12

Experimentally, the first excited state in ^{12}C is a 2^+ state, just as in ^{8}Be above, at an excitation energy of 4.44 MeV, and the lowest 4^+ state is at 14.08 MeV, so the ratio of excitation energies, $E_x(4)/E_x(2) = 3.17$, is quite close to what one would expect based on a rotational model for these states. A difference with ^{8}Be is that in ^{12}C there are several other excited states between the lowest 2^+ state and the lowest 4^+ state. Here, however, we only concentrate on these two excited states.

With the SRG evolved chiral N^{3}LO 2NF potential the lowest 2^+ and 4^+ states are very well converged [33], with excitation energies of 3.0 MeV and 10.7 MeV respectively; see the left panel of figure 7. Again, we estimate these results to be converged to within a few percent, and they are basically independent of the SRG parameter α. The ratio of the excitation energies is $E_x(4)/E_x(2) = 3.6$, about 8% above the rotational value, and more than 10% above the

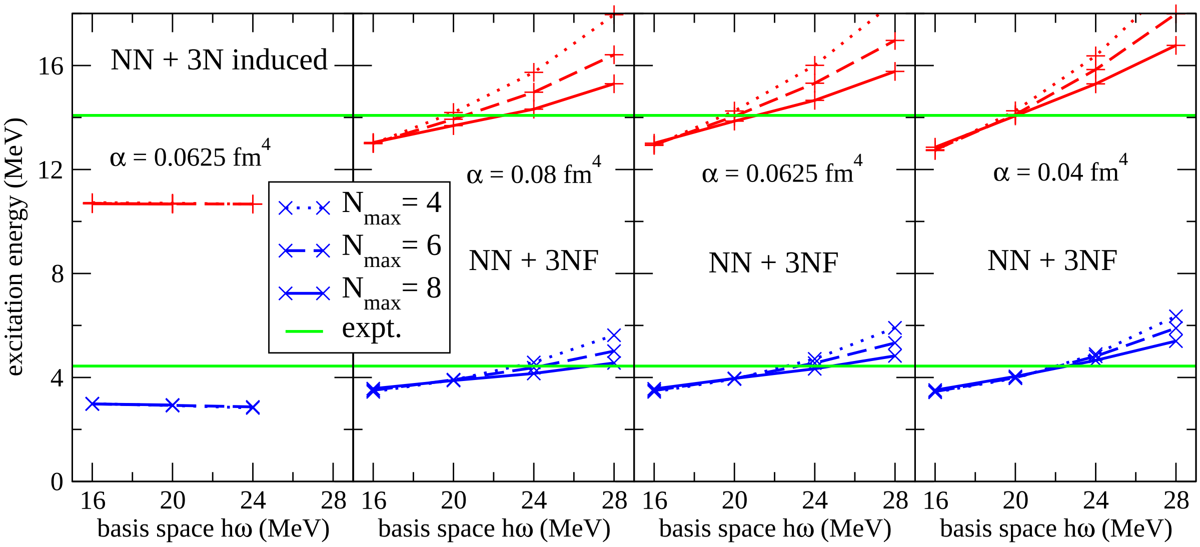

Figure 7. Excitation energies of the 2^+ (blue crosses) and 4^+ states (red plusses) for ^{12}C with SRG evolved chiral N^{3}LO 2NF plus induced 3NF at $\alpha = 0.0625$ fm^4 (left-most panel) and with SRG evolved chiral N^{3}LO 2NF plus chiral N^{2}LO 3NF. Experimental values are indicated by the horizontal green lines.

experimental value. Furthermore, the actual excitation energies are 25% to 30% below the data; also, there are no additional excited states below the lowest 4^+ state with the chiral N^{3}LO 2NF potential (at least no positive parity excited states), in contrast to the experimental situation.

However, in this case adding the chiral N^{2}LO 3NF to the N^{3}LO 2NF potential changes the results drastically; see the right three panels of figure 7. The initial chiral 3NF not only changes the excitation energies (and, for that matter, the spectrum in general), but it also spoils the excellent convergence of the excitation energies that we found with only the induced 3NFs. Though the weak dependence on $N_{\max}$ suggests that the excitation energies are almost converged at $\hbar\omega = 16$ MeV or $\hbar\omega = 20$ MeV, both the 2^+ and 4^+ states show a strong dependence on the basis space $\hbar\omega$ parameter: as $\hbar\omega$ increases from 16 MeV to 28 MeV, the excitation energies increase by 20% to 50%, depending on the state and on the SRG parameter α. It does mean that our results are now in better agreement with the data, but that is partly because of the rather large numerical uncertainty due to the lack of convergence. As expected, the convergence is somewhat better for the larger SRG value $\alpha = 0.08$ fm^4, but even at this SRG value we do not have converged results for the excitation energies. It appears that we need to go to larger bases ($N_{\max} = 10$ or even more) and/or larger values of α in order to obtain better converged results. With additional truncations on the basis space, such as the Importance-Truncated No-Core Shell Model [34], we can already perform calculations with 3NFs up to $N_{\max} = 12$ [26]. Alternatively, the Symmetry-Adapted No-Core Shell Model [35] may be used which aims to expand the range of accessible $N_{\max}$ values by implementing a truncation retaining those SU(3) configurations that are important for low-lying collective states.

The question remains: are these two states rotational excitations of the ground state? Without the initial chiral 3NF, the ratio of the excitation energies is about 3.6, that is, about 10% above the rotational value; see the top-left panel of figure 8. The ratio of the quadrupole moments, $Q(4)/Q(2)$ is very well converged at 1.3, only a few percent above the rotational value. Furthermore, the B(E2) transition strength for the transition from the 2^+ state to the ground state agrees to within 10% of a rotational model.

When we include the initial chiral N^{2}LO 3NF, the convergence patterns change, not only for

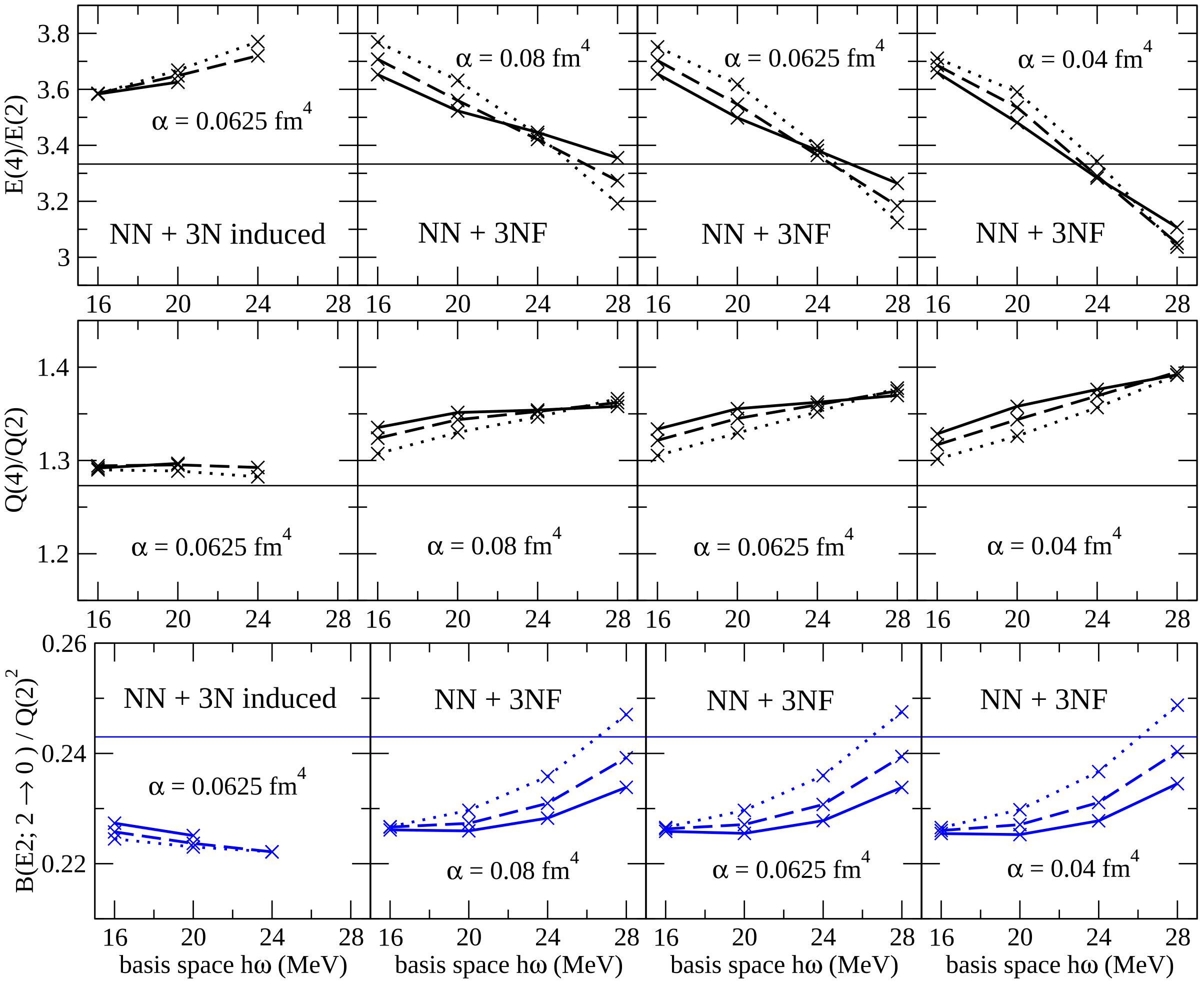

Figure 8. Ratios of results for ^{12}C with SRG evolved chiral N^{3}LO 2NF plus induced 3NF at $\alpha = 0.0625$ fm^4 (left-most panel) and with SRG evolved chiral N^{3}LO 2NF plus chiral N^{2}LO 3NF. Horizontal lines represent the predictions from a rotational model. See figure 7 for the legend of the line styles. Top: excitation energies $E_x(4)/E_x(2)$; Middle: quadrupole moments $Q(4)/Q(2)$; Bottom: E2 transition strength B(E2)$/Q(2)^2$ for the $2^+ \to 0^+$ transition.

the excitation energies, but also for the other observables. Although the excitation energies of these two states increases by as much as 50% as $\hbar\omega$ increases from 16 MeV to 28 MeV, the ratio $E_x(4)/E_x(2)$ changes by at most 20% over this range in $\hbar\omega$, and at $\alpha = 0.08$ fm^4 this ratio seems to converge to a value of about 3.5 ± 0.1, which is only 5% above the rotational value. Note that this is quite similar to the result we found without initial 3NFs, but the $\hbar\omega$ dependence is very different with and without initial 3NFs. The ratios $Q(4)/Q(2)$ and B(E2)$/Q(2)^2$ seem to converge to within about 10% of their rotational values; see figure 8. The convergence rate is slightly better at $\alpha = 0.08$ fm^4 than at $\alpha = 0.04$ fm^4, as one would expect.

Thus it seems that the rotational nature of the lowest 2^+ and the 4^+ states for ^{12}C emerges from ab initio calculations using chiral interactions. The chiral 3NFs do have a significant effect on the convergence rate of the excitation energies and on the spectrum of ^{12}C in general, as well as on other observables. However, the ratios of excitation energies, quadrupole moments, and B(E2)$/Q(J)^2$ transition strength seem to converge to within 10% of their rotational values, both with the chiral N^{3}LO 2NF only and with the chiral N^{3}LO 2NF plus N^{2}LO 3NF.

Acknowledgments

This work was supported in part by U.S. Department of Energy Grants DE-FC02-09ER41582 (SciDAC-2/UNEDF), DESC0008485 (SciDAC-3/NUCLEI), DE-FG02-87ER40371, DE-FC02-06ER2775, by the U.S. NSF Grants 0904782, CNS-0643969, OCI-0904809, OCI-0904802, by U.S. Department of Energy Contract DE-AC02-05CH11231, by GAUSTEQ (Germany and U.S. Nuclear Theory Exchange Program for QCD Studies of Hadrons and Nuclei) under contract number DE-SC0006758, and by the DFG through SFB 634 and by HIC for FAIR. Computational resources were provided by the National Energy Research Scientific Computing Center (NERSC), which is supported by the DOE Office of Science, and by an INCITE award, "Nuclear Structure and Nuclear Reactions", from the DOE Office of Advanced Scientific Computing. This research also used resources of the Oak Ridge Leadership Computing Facility at ORNL, which is supported by the DOE Office of Science under Contract DE-AC05-00OR22725.

References

[1] Wiringa R B, Stoks V G J and Schiavilla R 1995 *Phys. Rev.* **C51** 38

[2] Entem D R and Machleidt R 2003 *Phys. Rev.* **C68** 041001

[3] Machleidt R and Entem D R 2011 *Phys. Rept.* **503** 1

[4] Pieper S C, Pandharipande V R, Wiringa R B and Carlson J 2001 *Phys. Rev.* **C64** 014001

[5] Epelbaum E, Nogga A, Gloeckle W, Kamada H, Meissner U G and Witala H 2002 *Phys. Rev.* **C66** 064001

[6] Epelbaum E 2006 *Prog. Part. Nucl. Phys.* **57** 654

[7] Nogga A, Navrátil P, Barrett B and Vary J P 2006 *Phys. Rev.* **C73** 064002

[8] Navrátil P, Gueorguiev V G, Vary J P, Ormand W E and Nogga A 2007 *Phys. Rev. Lett.* **99** 042501

[9] Navrátil P, Vary J P and Barrett B R 2000 *Phys. Rev. Lett.* **84** 5728

[10] Caprio M A, Maris P and Vary J P 2012 *Phys. Rev.* **C86** 034312

[11] Abe T, Maris P, Otsuka T, Shimizu N, Utsuno Y and Vary J P 2012 *Phys. Rev.* **C86** 054301

[12] Maris P 2012 *J. Phys. Conf. Ser.* **402** 012031

[13] Vary J P, Maris P, Ng E, Yang C and Sosonkina M 2009 *J. Phys. Conf. Ser.* **180** 012083

[14] Sternberg P, Ng E G, Yang C, Maris P, Vary J P, Sosonkina M and Le H V 2008 in *Proc. of the 2008 ACM/IEEE conf. on Supercomputing* SC '08 (Piscataway, NJ: IEEE Press) 15:1

[15] Maris P, Sosonkina M, Vary J P, Ng E G and Yang C 2010 *Procedia Computer Science* **1** 97

[16] Aktulga H M, Yang C, Ng E G, Maris P and Vary J P 2012 in *Lecture Notes in Computer Science* 7484 (Springer) 830

[17] Aktulga H M, Yang C, Ng E G, Maris P and Vary J P 2012 Improving the Scalability of a Symmetric Iterative Eigensolver for Multi-core Platforms, submitted to *Concurrency and Computation: Practice and Experience*

[18] Roth R, Calci A, Langhammer J and Binder S 2013 (in preparation)

[19] Glazek S D and Wilson K G 1993 *Phys. Rev.* **D48** 5863

[20] Wegner F J 2001 *Physics Reports* **348** 77

[21] Bogner S, Furnstahl R and Perry R 2007 *Phys. Rev.* **C75** 061001

[22] Bogner S K *et al.* 2008 *Nucl. Phys.* **A801** 21

[23] Roth R, Neff T and Feldmeier H 2010 *Prog. Part. Nucl. Phys.* **65** 50

[24] Jurgenson E D, Navrátil P and Furnstahl R J 2009 *Phys. Rev. Lett.* **103** 082501

[25] Jurgenson E D, Navrátil P and Furnstahl R J 2011 *Phys. Rev.* **C83** 034301

[26] Roth R, Langhammer J, Calci A, Binder S and Navrátil P 2011 *Phys. Rev. Lett.* **107** 072501

[27] Jurgenson E D, Maris P, Furnstahl R J, Navrátil P, Ormand W E and Vary J P 2013 *Phys. Rev.* **C87** 054312

[28] Furnstahl R, Hagen G and Papenbrock T 2012 *Phys. Rev.* **C86** 031301

[29] Rowe D J 1970 *Nuclear Collective Motion* (London: Methuen, London)

[30] Caprio M A, Maris P and Vary J P 2013 *Phys. Lett. B* **719** 179

[31] Navrátil P 2007 *Few Body Syst.* **41** 117

[32] Gazit D, Quaglioni S and Navrátil P 2009 *Phys. Rev. Lett.* **103** 102502

[33] Maris P *et al* 2012 *J. Phys. Conf. Ser.* **403** 012019

[34] Roth R 2009 *Phys. Rev.* **C79** 064324

[35] Dytrych T *et al* 2013 (in preparation)