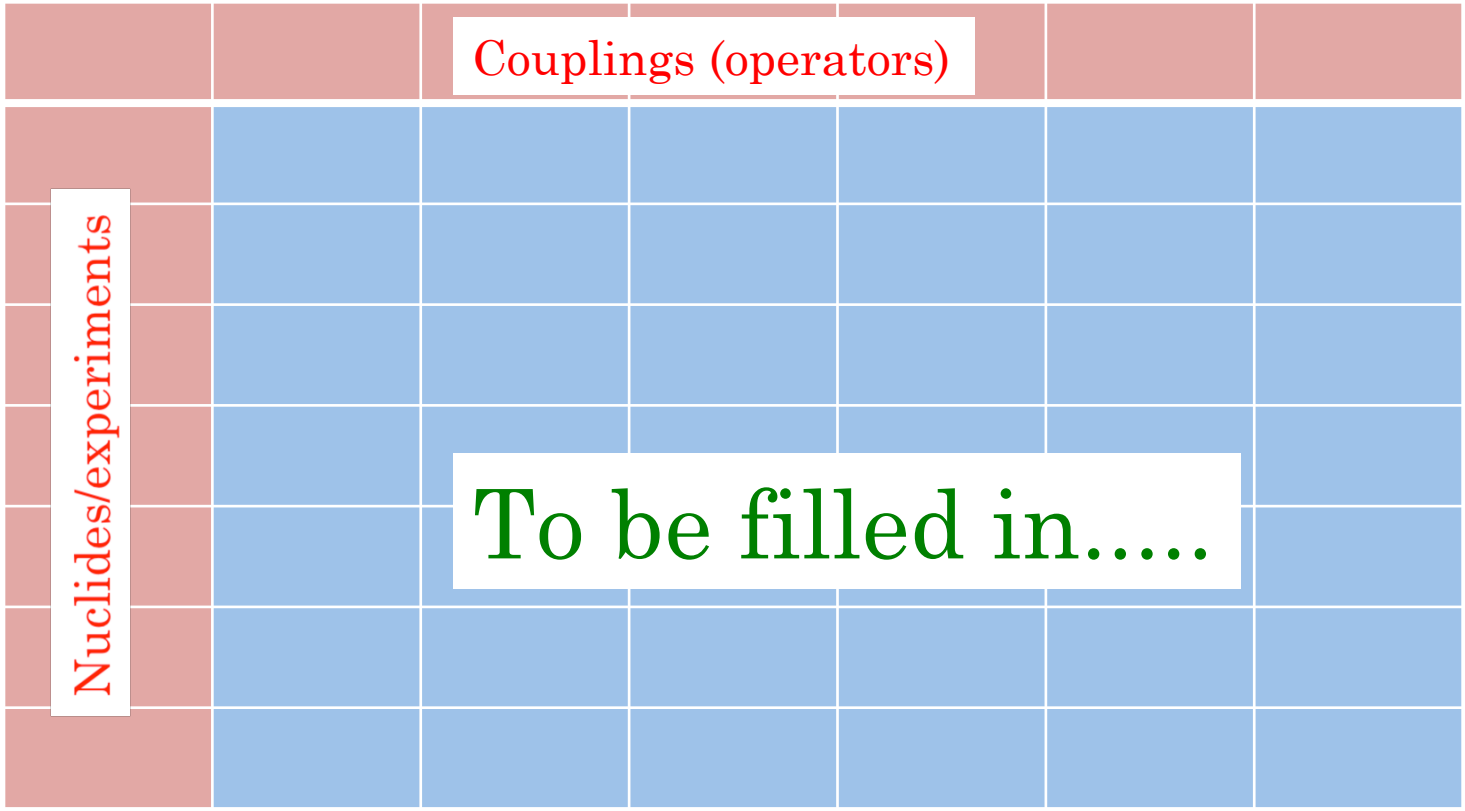
Nuclear structure calculations for an atlas of dark matter experiments

Calvin W. Johnson, San Diego State University

Collaborators:
Wick Haxton, UC Berkeley/LBL
Ken McElvain, UC Berkeley/ LBL

Other BIGSTICK collaborators
W. Erich Ormand, LLNL
Plamen G. Krastev, SDSU/Harvard
Hai Ah Nam, SDSU/ ORNL
Michael Kruse, LLNL
Micah Schuster, SDSU

An atlas of dark matter experiments



Potential target nuclides:

$^{12,13}C$

$^{16,18}O$

^{19}F

$^{20,21,22}Ne$

^{23}Na

$^{28,29,30}Si$

$^{32,33,34}S$

$^{36,40}Ar$

$^{40,42,43,44,48}Ca$

$^{70,72,73,74,76}Ge$

$^{122,123,124,125,126,128,130}Te$

^{127}I

$^{128,129,130,131,132,134,136}Xe$

^{133}Cs

$^{180,182,183,184,186}W$

EXECUTIVE SUMMARY ON THE BIGSTICK CODE

A brief glossary

Configuration-interaction: Diagonalizing the many-body Hamiltonian in an occupation-space basis (Slater determinants)

M-scheme: Using a basis with fixed total M=Jz

J-scheme: Using a basis with fixed total J (each state a linear combination of M-scheme states)

No-core shell model (NCSM): All particles active; usually used with *ab initio* interaction, h.o. single-particle states, often (not always) with N_{max} truncation.

Jumps: Instead of storing matrix elements, arrays which encode $< f \mid a^+ a \mid i >$, $< f \mid a^+ a^+ a a \mid i >$ etc. for calculating matrix elements on-the-fly

EXECUTIVE SUMMARY ON THE BIGSTICK CODE

Many-fermion code: 2nd generation after REDSTICK code
(started in *Baton Rouge, La.*)

Uses "factorization" algorithm: Johnson, Ormand, and Krastev,
Comp. Phys. Comm. **184**, 2761(2013)

Arbitrary single-particle radial waveforms
Allows local or nonlocal two-body interaction
Three-body forces implemented and validated
Applies to both nuclear and atomic cases

Runs on both desktop and parallel machines
--can run at least dimension 200M+ on desktop
--has done dimension 2 billion+ on supercomputers

Inline calculations of one-body density matrices, single-particle occupations,
(+ options to compute strength functions via Lanczos trick, etc.)

45 kilolines of code
Fortran 90 + MPI + OpenMP

EXECUTIVE SUMMARY ON THE BIGSTICK CODE

Why configuration-interaction? Why not other many-body methods?

Green's-function Monte Carlo:
Highly successful *ab initio* method, works well with "hard" NN interactions
Currently requires local interactions; tops out at around A=12-16;
excited states are difficult

Coupled cluster:
Highly successful *ab initio* method; widely applied in AMO;
works well with "hard" NN interactions; can handle heavier nuclei;
good scaling with number of particles/orbits (only linked diagrams)
Excited states are difficult; works best for closed shell, closed shell +/- 1
Cannot do N_{max} truncation

Configuration interaction
Generates excited states easily; works well with mid-shell nuclides
Computational efforts increases exponentially with # particles, orbits
due to "unlinked diagrams"; thus requires a "soft" interaction
which does not connect high and low momentum states

EXECUTIVE SUMMARY ON THE BIGSTICK CODE

BIGSTICK has run on a number of platforms

It runs easily and routinely on laptops and desktop Linux boxes and Macs.

It has run successfully on large parallel systems:

- Jaguar PF (ORNL) (MPI) up to about 5000 MPI processes
- Sierra (LLNL) including in hybrid MPI-OpenMP up to 1000 processes
- Franklin, Hopper, and Edison (NERSC) (MPI) up to 10,000 MPI procs
- Stampede (TACC/XSEDE machine) (MPI+OpenMP) up to 200 MPI procs

The most recent versions successfully break up the Lanczos vectors: the biggest dimension achieved so far is over 2 billion basis states (however, the computational barriers are more complex than just the dimension of the vectors—more on that in a moment.)

THE BASIC PROBLEM

The basic *science question* is to model detailed quantum structure of many-body systems, such the structure of an atomic nucleus.

The algorithms described today are best applied to many body systems with
(a)two "species" (protons and neutrons, or +1/2 and -1/2 electrons)
(b)single-particle basis states with good rotational symmetry (j, m)

To answer this, we solve *Schrödinger's equation*:

$$\hat{\mathbf{H}}|\Psi\rangle = E|\Psi\rangle$$

* **H** is generally a very large matrix – dimensions up to 10^{10} have been tackled.
* **H** is generally very sparse.
* We usually only want a few low-lying states

THE KEY IDEAS

Basic problem: find extremal eigenvalues of very large, very sparse Hermitian matrix

Lanczos algorithm

fundamental operation is *matrix-vector multiply*

Despite sparsity, nonzero matrix elements can require TB of storage

Only a fraction of matrix elements are unique; **most are reused.**
Reuse of matrix elements understood through *spectator* particles.

Reuse can be **exploited using exact factorization**
enforced through *additive/multiplicative quantum numbers*

THE BASIC PROBLEM

Find extremal eigenvalues of very large, very sparse Hermitian matrix

→ Lanczos algorithm

fundamental operation is *matrix-vector multiply*

$$|\Psi\rangle = \sum_{\alpha} c_{\alpha} |\alpha\rangle \qquad H_{\alpha\beta} = \langle \alpha | \hat{\mathbf{H}} | \beta \rangle$$

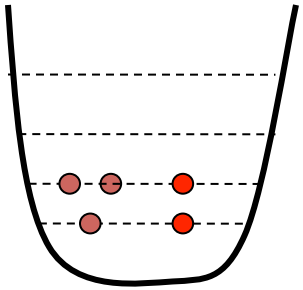
$$\sum_{\beta} H_{\alpha\beta} c_{\beta} = E c_{\alpha} \quad \text{if} \quad \langle \alpha | \beta \rangle = \delta_{\alpha\beta}$$

so we use the matrix formalism

$$\hat{\mathbf{H}} |\Psi\rangle = E |\Psi\rangle$$

Nuclear Hamiltonian: $\hat{H} = \sum_i -\frac{\hbar^2}{2M}\nabla_i^2 + \sum_{i<j} V(r_i, r_j)$

Solve by diagonalizing **H** in a basis of many-body states.
The many-body states are *Slater determinants*, or
anti-symmeterized products of single-particle wfns.



The single-particle states are defined by a single-particle potential *U(r)* (such as harmonic oscillator or Hartree-Fock)

Maria Mayer

At this point one generally goes to occupation representation:

$$\hat{H} = \sum_i \varepsilon_i \hat{a}_i^+ \hat{a}_i + \frac{1}{4}\sum_{ijkl} V_{ijkl} \hat{a}_i^+ \hat{a}_j^+ \hat{a}_l \hat{a}_k$$

single-particle energies

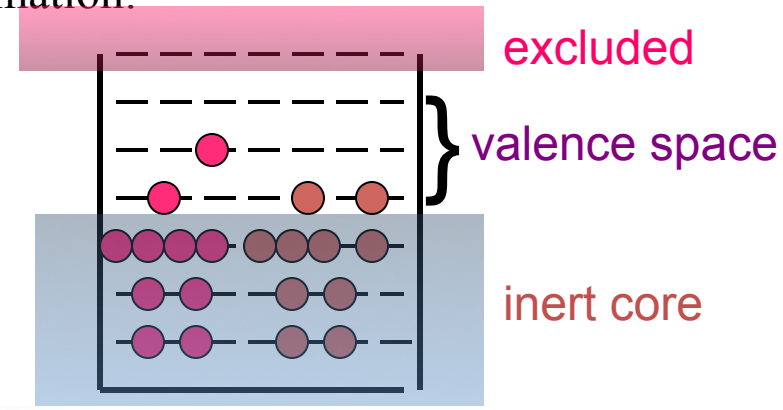
two-body matrix elements

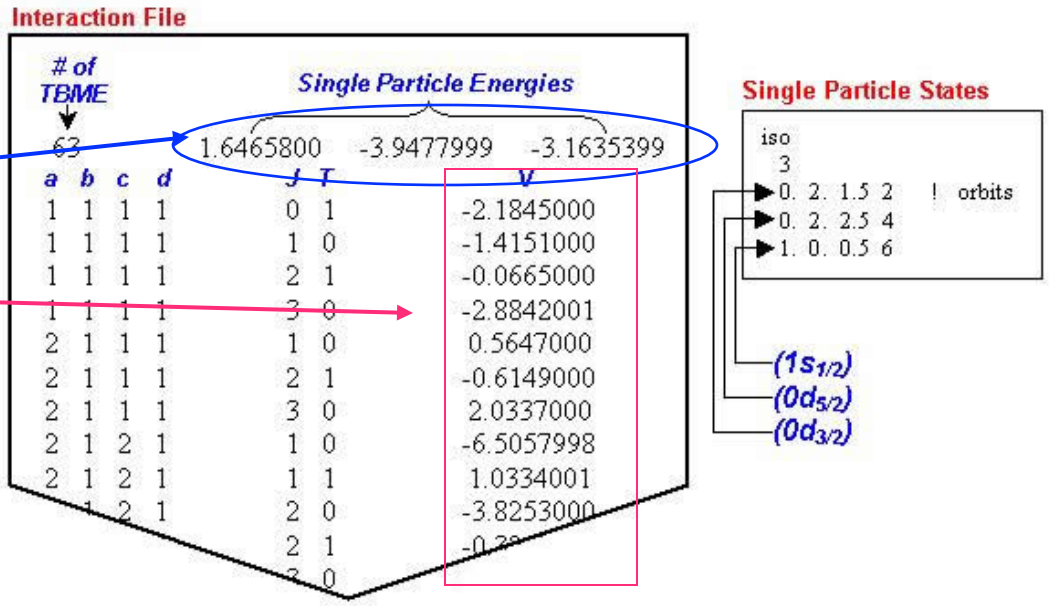
When running a fermion shell model code (e.g. MFD, BIGSTICK), one enters the following information:

(1) The single-particle valence space (such as *sd* or *pf*); assumes inert core

(2) The many-body model space (number of protons and neutrons, truncations, etc.)

(3) The interaction:
single-particle energies
and
two-body matrix elements
$V_{JT}(ab,cd)$





A SPARSE MATRIX, BUT....

Despite sparsity, nonzero matrix elements can require TB of storage

- How the basis states are represented

Product wavefunction ("Slater Determinant")

$$\Psi(\vec{r}_1,\vec{r}_2,\vec{r}_3\ldots) = \phi_{n_1}(\vec{r}_1)\phi_{n_2}(\vec{r}_2)\phi_{n_3}(\vec{r}_3)\ldots\phi_{n_N}(\vec{r}_N)$$

Each many-body state can be *uniquely* determined by a list of "occupied" single-particle states = "occupation representation"

$$|\alpha\rangle = \hat{a}^+_{n_1}\hat{a}^+_{n_2}\hat{a}^+_{n_3}\ldots\hat{a}^+_{n_N}|0\rangle$$

A SPARSE MATRIX, BUT....

Despite sparsity, nonzero matrix elements can require TB of storage

- How the basis states are represented

"occupation representation" $\left|\alpha\right\rangle = \hat{a}^{+}_{n_1}\hat{a}^{+}_{n_2}\hat{a}^{+}_{n_3}\ldots\hat{a}^{+}_{n_N}\left|0\right\rangle$

n_i	1	2	3	4	5	6	7
α=1	1	0	0	1	1	0	1
α=2	1	0	1	0	0	1	1
α=3	0	1	1	1	0	1	0

A SPARSE MATRIX, BUT....

Despite sparsity, nonzero matrix elements can require TB of storage

- How the basis states are represented

some technical details:
the "M-scheme"

$$|\alpha\rangle = \hat{a}^{+}_{n_1} \hat{a}^{+}_{n_2} \hat{a}^{+}_{n_3} \ldots \hat{a}^{+}_{n_N} |0\rangle$$

label	N	l	m_l
1	1	0 (S)	0
2	2	0 (S)	0
3	2	1 (P)	1
4	2	1 (P)	0
5	2	1 (P)	-1

For any Slater determinant, the total M = sum of the m_l's

Because J_z commutes with H, we can use a basis with M fixed = "M-scheme"

A SPARSE MATRIX, BUT....

Despite sparsity, nonzero matrix elements can require TB of storage

- Typical dimensions and sparsity

Nuclide	valence space	valence Z	valence N	basis dim	sparsity (%)
^{20}Ne	"sd"	2	2	640	10
^{25}Mg	"sd"	4	5	44,133	0.5
^{49}Cr	"pf"	4	5	6M	0.01
^{56}Fe	"pf"	6	10	500M	2×10^{-4}

A SPARSE MATRIX, BUT....

Despite sparsity, nonzero matrix elements can require TB of storage

Nuclide	Space	Basis dim	matrix store
^{56}Fe	*pf*	501 M	4.2 Tb
^{7}Li	N_{max}=12	252 M	3.6 Tb
^{7}Li	N_{max}=14	1200 M	23 Tb
^{12}C	N_{max}=6	32M	0.2 Tb
^{12}C	N_{max}=8	590M	5 Tb
^{12}C	N_{max}=10	7800M	111 Tb
^{16}O	N_{max}=6	26 M	0.14 Tb
^{16}O	N_{max}=8	990 M	9.7 Tb

Recycled Matrix Elements

Only a fraction of matrix elements are unique; **most are reused.**

Reuse of matrix elements understood through *spectator* particles.

# of nonzero matrix elements vs. # unique matrix elements

Nuclide	valence space	valence Z	valence N	# nonzero	# unique
^{28}Si	"sd"	6	6	26 x 10^6	3600
^{52}Fe	"pf"	6	6	90 x 10^9	21,500

FACTORIZATION

Reuse can be **exploited using exact factorization**
enforced through *additive/multiplicative quantum numbers*

We work in an *M*-scheme basis:

Because $\mathbf{J}^2$ and $\mathbf{J}_z$ both commute with **H**, one does not need *all* basis states, but can use many-body basis restricted to the same *M*.

This is easy because *M* is an additive quantum number so it is possible for a single Slater determinant to be a state of good *M*.

(It's possible to work in a *J*-basis, e.g. OXBASH or NuShell, but each basis state is generally a complicated sum of Slater determinants).

FACTORIZATION

Reuse can be **exploited using exact factorization**
enforced through *additive/multiplicative quantum numbers*

Because the M values are discrete integers or half-integers
(-3, -2, -1, 0, 1, 2, … or -3/2, -1/2, +1/2, +3/2….)
we can organize the basis states in discrete *sectors*

Example: 2 protons, 4 neutrons, total M = 0

$M_z(\pi) = -4$	$M_z(\upsilon) = +4$
$M_z(\pi) = -3$	$M_z(\upsilon) = +3$
$M_z(\pi) = -2$	$M_z(\upsilon) = +2$

FACTORIZATION

Reuse can be **exploited using exact factorization**
enforced through *additive/multiplicative quantum numbers*

In fact, we can see an example of factorization here because all proton Slater determinants in one M-sector *must* combine with all the conjugate neutron Slater determinants

Example: 2 protons, 4 neutrons, total M = 0

$M_z(\pi) = -4$: 2 SDs	$M_z(\upsilon) = +4$: 24 SDs	48 combined
$M_z(\pi) = -3$: 4 SDs	$M_z(\upsilon) = +3$: 39 SDs	156 combined
$M_z(\pi) = -2$: 9 SDs	$M_z(\upsilon) = +2$: 60 SDs	540 combined

FACTORIZATION

Reuse can be **exploited using exact factorization**
enforced through *additive/multiplicative quantum numbers*

In fact, we can see an example of factorization here because all proton Slater determinants in one M-sector *must* combine with all the conjugate neutron Slater determinants

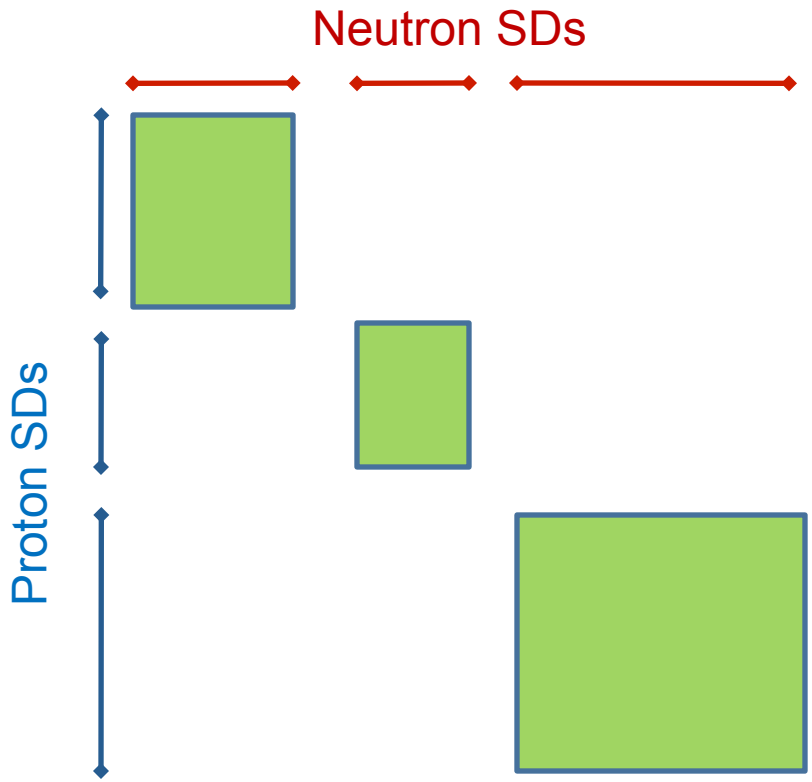
$M_z(\pi) = -4$: 2 SDs	$M_z(\upsilon) = +4$: 24 SDs	48 combined

$$\begin{matrix} |\pi_1\rangle \\ |\pi_2\rangle \end{matrix} \times \begin{matrix} |\nu_1\rangle \\ |\nu_2\rangle \\ |\nu_3\rangle \\ |\nu_4\rangle \\ \vdots \\ |\nu_{24}\rangle \end{matrix} = \begin{matrix} |\pi_1\rangle|\nu_1\rangle \\ |\pi_2\rangle|\nu_1\rangle \\ |\pi_1\rangle|\nu_2\rangle \\ |\pi_2\rangle|\nu_2\rangle \\ \vdots \\ |\pi_1\rangle|\nu_{24}\rangle \\ |\pi_2\rangle|\nu_{24}\rangle \end{matrix}$$

FACTORIZATION

Reuse can be **exploited using exact factorization**
enforced through *additive/multiplicative quantum numbers*

$$|\alpha\rangle = |\alpha_p\rangle \times |\alpha_n\rangle$$



Example N = Z nuclei

Nuclide	Basis dim	# pSDs (=#nSDs)
^{20}Ne	640	66
^{24}Mg	28,503	495
^{28}Si	93,710	924
^{48}Cr	1,963,461	4895
^{52}Fe	109,954,620	38,760
^{56}Ni	1,087,455,228	125,970

FACTORIZATION

Reuse can be **exploited using exact factorization**
enforced through *additive/multiplicative quantum numbers*

Factorization allows us to keep track of all basis states
without writing out every one explicitly
-- we only need to write down the proton/neutron components

The same trick can be applied to matrix-vector multiply

$$\hat{H} = \hat{H}_{pp} + \hat{H}_{nn} + \hat{H}_{pn}$$

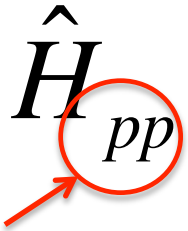
Move 2 protons; *neutrons are spectators*

Move 2 neutrons; *protons are spectators*

Move 1 proton + 1 neutron; *rest are spectators*

FACTORIZATION

Reuse can be **exploited using exact factorization**
enforced through *additive/multiplicative quantum numbers*



Move 2 protons;
neutrons are spectators

Example: 2 protons, 4 neutrons, total M = 0

$M_z(\pi) = -4$: 2 SDs	$M_z(\upsilon) = +4$: 24 SDs	48 combined

There are potentially 48 × 48 matrix elements
But for H_{pp} at most 4 × 24 are nonzero
and we only have to look up 4 matrix elements

Advantage: **we can store 98 matrix elements as 4 matrix elements** and avoid 2000+ zero matrix elements.

FACTORIZATION

Reuse can be **exploited using exact factorization**
enforced through *additive/multiplicative quantum numbers*

$M_z(\pi) = -4$: 2 SDs	$M_z(\upsilon) = +4$: 24 SDs	48 combined

$$|\pi_1\rangle \quad |\pi_2\rangle \qquad H_{pp} = \begin{pmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{pmatrix}$$

$$|\nu_1\rangle \quad |\nu_2\rangle \quad |\nu_3\rangle \quad |\nu_4\rangle \quad \vdots \quad |\nu_{24}\rangle$$

$$H_{pp}|\pi_1\rangle|\nu_1\rangle = H_{11}|\pi_1\rangle|\nu_1\rangle + H_{12}|\pi_2\rangle|\nu_1\rangle$$
$$H_{pp}|\pi_2\rangle|\nu_1\rangle = H_{12}|\pi_1\rangle|\nu_1\rangle + H_{22}|\pi_2\rangle|\nu_1\rangle$$
$$H_{pp}|\pi_1\rangle|\nu_2\rangle = H_{11}|\pi_1\rangle|\nu_2\rangle + H_{12}|\pi_2\rangle|\nu_2\rangle$$
$$H_{pp}|\pi_2\rangle|\nu_2\rangle = H_{12}|\pi_1\rangle|\nu_2\rangle + H_{22}|\pi_2\rangle|\nu_2\rangle$$
$$\vdots$$
$$H_{pp}|\pi_1\rangle|\nu_{24}\rangle = H_{11}|\pi_1\rangle|\nu_{24}\rangle + H_{12}|\pi_2\rangle|\nu_{24}\rangle$$
$$H_{pp}|\pi_2\rangle|\nu_{24}\rangle = H_{12}|\pi_1\rangle|\nu_{24}\rangle + H_{22}|\pi_2\rangle|\nu_{24}\rangle$$

Advantage: **we can store 98 matrix elements as 4 matrix elements** and avoid 2000+ zero matrix elements.

FACTORIZATION

Reuse can be **exploited using exact factorization**
enforced through *additive/multiplicative quantum numbers*

Comparison of nonzero matrix storage with factorization

Nuclide	Space	Basis dim	matrix store	factorization
^{56}Fe	*pf*	501 M	290 Gb	0.72 Gb
^{7}Li	N_{max}=12	252 M	3600 Gb	96 Gb
^{7}Li	N_{max}=14	1200 M	23 Tb	624 Gb
^{12}C	N_{max}=6	32M	196 Gb	3.3 Gb
^{12}C	N_{max}=8	590M	5000 Gb	65 Gb
^{12}C	N_{max}=10	7800M	111 Tb	1.4 Tb
^{16}O	N_{max}=6	26 M	142 Gb	3.0 Gb
^{16}O	N_{max}=8	990 M	9700 Gb	130 Gb

"It's not enough to just show up. You have to have a business plan."

Potential target nuclides and M-scheme dimensionalities

^{12,13}C

^{16,18}O

^{19}F

^{20,21,22}Ne

^{23}Na

^{28,29,30}Si

^{32,33,34}S

"easy"
with standard
spaces and interactions

^{36,40}Ar

^{40,42,43,44,48}Ca

^{70,72,73,74,76}Ge

^{122,123,124,125,126,128,130}Te

^{127}I

^{128,129,130,131,132,134,136}Xe

^{133}Cs

^{180,182,183,184,186}W

Potential target nuclides and M-scheme dimensionalities

Single-particle space: $1s_{1/2}\ 0d_{3/2}\ 0f_{7/2}\ 1p_{3/2}\ 1p_{1/2}$

Nuclide	dimension
^{36}Ar	1.1 million
^{40}Ar	25 million
^{40}Ca	60 M
^{42}Ca	190 M
^{43}Ca	250 M
^{44}Ca	276 M
^{48}Ca	60 M

Comment: Moderately easy but need to tune interaction

Potential target nuclides and M-scheme dimensionalities

Single-particle space: $1p_{3/2}\ 1p_{1/2}\ 0f_{5/2}\ 0g_{9/2}$

Nuclide	dimension
^{70}Ge	140 M
^{72}Ge	140 M
^{73}Ge	108 M
^{74}Ge	70 M
^{76}Ge	17 M

Comment: Moderately easy; JUN45 interaction mostly tuned

Potential target nuclides and M-scheme dimensionalities

Single-particle space: $2s_{1/2}\ 1d_{3/2}\ 1d_{5/2}\ 0g_{7/2}\ 0h_{11/2}$

Nuclide	dimension
^{122}Te	2.8 Billion
^{123}Te	1.6 B
^{124}Te	820 Million
^{125}Te	360 M
^{126}Te	141 M
^{128}Te	13 M
^{130}Te	0.6 M

Nuclide	dimension
^{128}Xe*	9 Billion
^{129}Xe	3 B
^{130}Xe	850 Million
^{131}Xe	198 M
^{132}Xe	3.7 M
^{134}Xe	0.6 M
^{136}Xe	1504

Nuclide	dimension
^{127}I	1.3 Billion
^{133}Cs	198 million

Comment: Challenging, interaction must be tuned

*Abundance = 1.91%; can truncate to 1.35 B

Potential target nuclides and M-scheme dimensionalities

Single-particle space: $0h_{11/2}\ 0h_{9/2}\ 1f_{7/2}\ 1f_{5/2}\ 2p_{3/2}\ 2p_{1/2}\ 0i_{11/2}$

Nuclide	dimension
^{180}W*	187 Billion
^{182}W	2.5 B
^{183}W	206 Million
^{184}W	13 M
^{186}W	12301

Comment: Very challenging; need interaction

*Abundance = 0.12%; can truncate