Defeating the curse of dimensionality to compute a vibrational spectrum

Gustavo Avila, Arnaud Leclerc, and Tucker Carrington

Chemistry Department
Queen's University

27 mars 2014

Objective

Compute a vibrational spectrum by solving the vibrational Schroedinger equation

$$\hat{H}\psi_k = E_k\psi_k$$

$$\hat{H} = \hat{K} + \hat{V} \ .$$

Represent wavefunctions with basis functions

$$\psi_k(\mathbf{q}) = \sum_n c_n^k \quad f_n(\mathbf{q})$$

General potentials

If the potential, $\hat{V}$, is general, i.e.,

- not as a sum-of-products
- not a sum of terms with one, two, etc coordinates

We have developed two methods :

- one with a Smolyak quadrature
- one based a Smolyak-inspired collocation

both these methods require storing vectors with as many components as basis functions.

Sum-of-products potentials

If the potential is a sum of products one can exploit its structure to avoid storing vectors with as many components as basis functions.

We have developed a reduced rank power method scheme.

The variational (Galerkin) method is common in chemistry

- represent wavefunctions with basis functions

$$\psi_k(\mathbf{q}) = \sum_n c_n^k \quad f_n(\mathbf{q})$$

- multiply on the left with $f_m(\mathbf{q})$, integrate to obtain a matrix eigenvalue problem
- compute eigenvalues and eigenvectors of the Hamiltonian matrix

Basis sets and quadrature grids are huge

Often one uses product basis functions :

$$f_{k_1,k_2,\cdots} = \phi_{k_1}(r_1)\phi_{k_2}(r_2)\cdots\phi_{k_N}(\theta_1)\cdots$$

Between 10 and 100 1-D functions required for each coordinate.

If n basis functions are required for each coordinate and there are D coordinates then the size of the basis is n^D.

To compute vibrational levels $> 10^{3N-6}$ multi-d basis functions are required.

Does this poor scaling matter?

$$n_{1d} \approx 10$$

H_2O
Size of matrix 10^3

CH_2O
Size of matrix 10^6

C_2H_4
Size of matrix 10^{12}

The curse of dimensionality

The variational method limits one's choices

- coordinates and (orthogonal) basis functions are chosen so that matrix elements of the kinetic energy operator (KEO) can be calculated exactly (analytically)
- a quadrature is chosen that is exact for all overlap matrix elements
- one solves $\mathbf{HU} = \mathbf{UE}$
    - there are no efficient iterative eigensolvers for $\mathbf{HU} = \mathbf{SUE}$

To make an effective variational method one must reduce the size of the basis and the quadrature grid

It is common to use product basis functions that are eigenfunctions of a zeroth-order Hamiltonian,

$$H = H_0 + \Delta$$

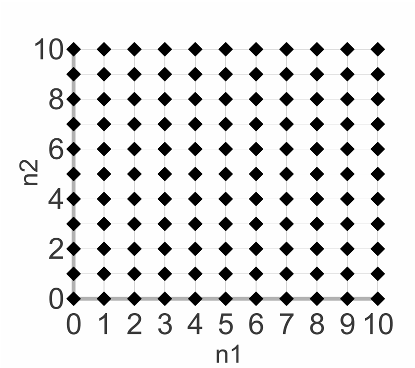
H_0 is a sum of 1d Hamiltonians (separable).

One can remove basis functions with large zeroth-order energies.

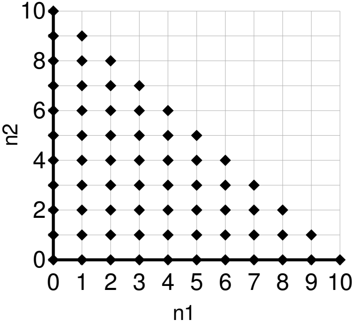
If all the 1d Hamiltonians are identical one simply removes basis functions for which

$$\sum_c n_c > b$$

Full basis for a 2d problem



Pruned basis for the 2d problem



If $3N - 6 = 15$ and 15 basis functions are used for each coordinate then the size of the direct product basis is 4×10^{17}.

By discarding all functions for which $\sum_c n_c > b = 15$ the size of the basis is reduced to 7.7×10^7.

$$\boxed{\text{Basis vector : } 3 \times 10^9 \ \text{ GB} \rightarrow 0.6 \text{ GB}}$$

It is also possible to reduce the size of the quadrature grid

- For a 12D problem, a direct product quadrature has $\sim 15^{12}$ points. Storing one vector requires about 10^6 GB.
- We must find a smaller grid with enough structure that we can efficiently evaluate matrix-vector products

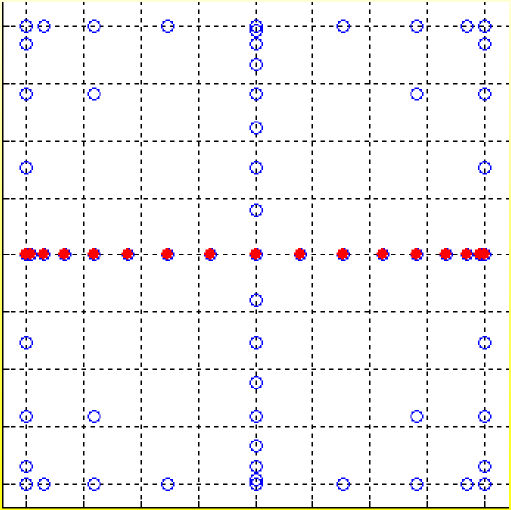
Smolyak grids

The Smolyak quadrature equation for integrating a function $g(x_1, x_2, \cdots, x_D)$ can be written as a sum of D-dimensional product quadrature grids,

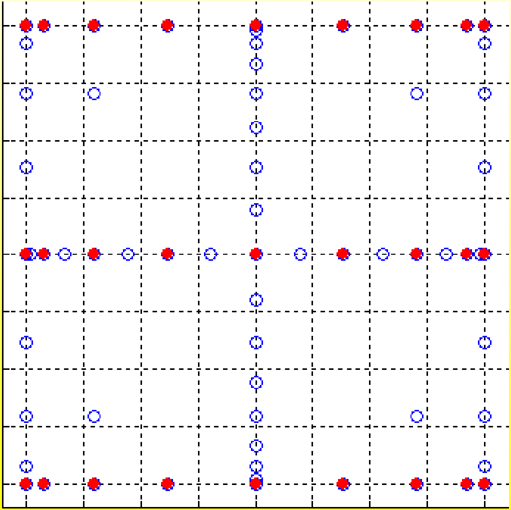
$$S(D,H) = \sum_{i_1+i_2+\cdots \leq H} C_{i_1,\ldots,i_D}[Q^{i_1}(x_1) \otimes \cdots \otimes Q^{i_D}(x_D)],$$

Using these ideas calculations for molecules with 6 atoms are possible.

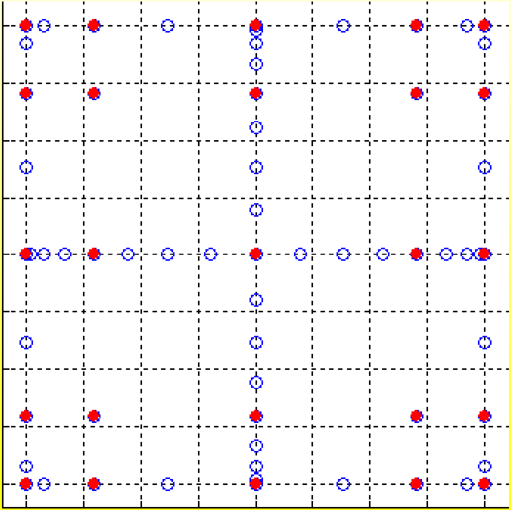
Smolyak Quadrature: 2D Level4 17x1 component



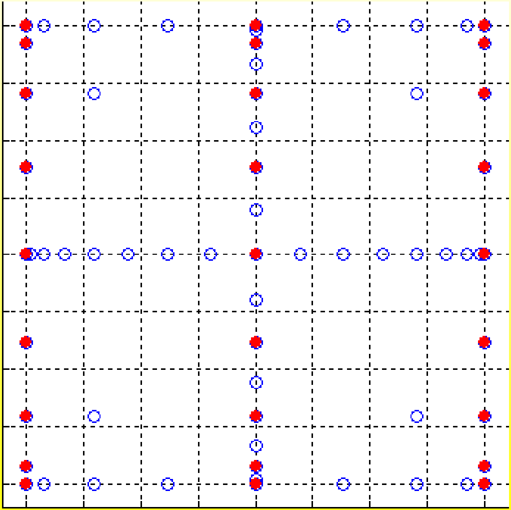
Smolyak Quadrature: 2D Level4 9x3 component



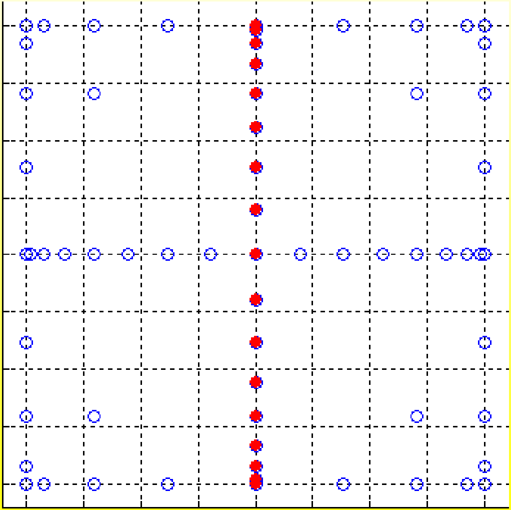
Smolyak Quadrature: 2D Level4 5x5 component



Smolyak Quadrature: 2D Level4 3x9 component



Smolyak Quadrature: 2D Level4 1x17 component



Smolyak quadratures adapted to our bases work well

Grid size $\sim 5.7 \times 10^{13} \rightarrow 8.5 \times 10^{6}$

memory cost 500 TB $\rightarrow$ 0.07 GB

Why bother with collocation ?

Collocation obviates

- integrals
- the need for basis functions with which matrix elements of the KEO are exact
- the need for orthogonal basis functions

What is collocation ?

Apply $\hat{H} - E_k$ to

$$\psi_k(\mathbf{q}) = \sum_n u_n^k \;\; f_n(\mathbf{q})$$

and determine the u_n^k by demanding that the Schroedinger equation be satisfied at a set of points, i.e. solve

$$(\mathbf{T} + \mathbf{VB})\mathbf{U} = \mathbf{BUE} \; ,$$

Two drawbacks of established collocation methods

- $\mathbf{B} \neq \mathbf{I}$

  and it is necessary to solve a generalized eigenvalue problem

- The **H** and **B** matrices of the collocation eigenvalue problem,

  $\mathbf{HU} = \mathbf{BUE}$

  are not symmetric

Overcoming the drawbacks

$\mathbf{B} \neq \mathbf{I}$ makes collocation almost unusable (when the number of basis functions is larger than about 50'000).
There are good tools for computing eigenvalues and eigenvectors of a nonsymmetric eigenvalue problem, if $\mathbf{B} = \mathbf{I}$

In this talk I present a new collocation method that obviates the need to solve a generalized eigenvalue problem

- A new tool for solving the Schroedinger equation.

Avoiding a generalized eigenvalue problem

In 1-D this is easily accomplished by using as basis functions Lagrange-like basis functions that are one at one of the collocation points and zero at all the others.

($\mathbf{B} = \mathbf{I}$ because $f_b(x_a) = \delta_{ab}$)

1-D Lagrange-like functions

Functions that spans the same space as the first m harmonic oscillator functions,

$$a_j(x) \;=\; \exp\left(\frac{-x^2+x_j^2}{2}\right) \prod_{\substack{i=1\\ i\neq j}}^{m} \left(\frac{x-x_i}{x_j-x_i}\right) ,$$

Functions that spans the same space as the first m 1-D wavefunctions, ϕ_q,

$$a_k^m(x) = \sum_q \phi_q(x) c_q^k \ ,$$

where $\sum_{q=1}^{m-1} M_{k'q} c_q^k = \delta_{k,k'}$ with $M_{k'q} = \phi_q(x_{k'})$.

In D dimensions

In D dimensions a generalized eigenvalue problem can be avoided by using a direct product Lagrange-like basis.

The usefulness of a direct-product basis is limited by the curse of dimensionality.

If 10 basis functions per coordinate are necessary, the size of a direct product basis is 10^{3N-6}. This makes calculations costly, even when the Hamiltonian matrix is not computed, stored, and diagonalized.

How can one **both** avoid the curse and avoid a generalized eigenvalue problem ?

Multidimensional collocation

The key idea is not to apply $\hat{H} - E_n$ to a basis representation of a wavefunction, but to a Smolyak or sparse-grid interpolant.

$$I(D,H)\Phi_n(x_1,x_2,\cdots,x_D) = \sum_{g(i_1,i_2,\cdots,i_D)\leq H} C_{i_1,i_2,\cdots,i_D}$$

$$\times \sum_{k_1=1}^{m_{i_1}} \sum_{k_2=1}^{m_{i_2}} \cdots \sum_{k_D=1}^{m_{i_D}} \Phi_n(x_{k_1}^{i_1}, x_{k_2}^{i_2}, \cdots, x_{k_D}^{i_D}) a_{k_1}^{i_1}(x_1) a_{k_2}^{i_2}(x_2) \cdots a_{k_D}^{i_D}(x_D).$$

$C_{i_1,i_2,\cdots,i_D}$ coefficients are the same as those used with Smolyak quadrature and ($\mathbf{B} = \mathbf{I}$).

Multidimensional collocation

We use

$$g(i_1, i_2, \cdots, i_D) = i_1 + i_2 + \cdots + i_D \ ,$$

but better choices exist.

Smolyak interpolation is usually used with piecewise-linear basis functions. Instead, we use Lagrange-like functions that span the same space as a set 1-D eigenfunctions.

When $D = 6$ and $K(= H - D + 1) = 11$,

$$R_{SDP} = \frac{N_{Smolyak}}{N_{direct\ \ product}} \approx 5 \times 10^{-3} \ .$$

The ratio decreases as D increases :
for $D = 10$ and $K = 11$,

$$R_{SDP} = \frac{2 \times 10^5}{11^{10}} \approx 7 \times 10^{-6} \ ;$$

for $D = 15$ and $K = 11$,

$$R_{SDP} = \frac{3 \times 10^6}{11^{15}} \approx 8 \times 10^{-10} \ .$$

$N_{Smolyak}$ does not scale exponentially with H or D.

Requiring that the Schroedinger equation be satisfied at a point on the Smolyak grid means imposing

$$\sum_{g(i_1,i_2,\cdots,i_D)\leq H} C_{i_1,i_2,\cdots,i_D}$$

$$\times \sum_{k_1=1}^{m_{i_1}} \sum_{k_2=1}^{m_{i_2}} \cdots \sum_{k_D=1}^{m_{i_D}} \hat{K}\Phi_n(x_{k_1}^{i_1}, x_{k_2}^{i_2}, \cdots, x_{k_D}^{i_D}) a_{k_1}^{i_1}(x_{k_1'}) a_{k_2}^{i_2}(x_{k_2'}) \cdots a_{k_D}^{i_D}(x_{k_D'})$$

$$+ V_{x_{k_1'}, x_{k_2'}, \cdots, x_{k_D'}} \Phi_n(x_{k_1'}, x_{k_2'}, \cdots, x_{k_D'}) = E_n \Phi_n(x_{k_1'}, x_{k_2'}, \cdots, x_{k_D'}) \ .$$

The matrix eigenvalue problem

- $$(\mathbf{T}+\mathbf{V})\mathbf{U}=\mathbf{U}\mathbf{E},$$
- We do not construct **T**.
- Elements of the eigenvectors are values of wavefunctions at points.
- As *H* is increased, diagonal elements of **E** and columns of **U** converge to exact energies and wavefunction values at the Smolyak grid points.
- The number of points on the Smolyak grid is equal to the number of basis functions.

Matrix-vector products

Potential matrix-vector products are trivial.

Matrix-vector products for the KEO are done term by term.

For each term, sums are evaluated sequentially, exploiting the structure of the Smolyak grid.

KEO

The KEO can always be written

$$\hat{K} = \sum_{i=1}^{D} W^{i}(x_1, \cdots, x_D)\frac{\partial}{\partial x_i} + \sum_{i=1}^{D}\sum_{j\leq i}^{D} W^{i,j}(x_1, \cdots, x_D)\frac{\partial}{\partial x_i}\frac{\partial}{\partial x_j} \ ,$$

- Cost of computing the spectrum does not depend on the form or complexity of the functions W^i and $W^{i,j}$

Test calculation

- Normal coordinates
- Full Watson KEO
- Cut eigenfunctions for 1-D bases

Results

Table: Energies computed with $b = 20$ and $H = 26$

sym	Assig	Exp	Luckhaus (b)	This work (c)	$b - c$
A_1	ground		5777.44	5777.446	-0.01
B_1	ν_4	1167.4	1171.09	1171.094	0.00
B_2	ν_6	1249.6	1252.89	1252.888	0.00
A_1	ν_3	1500.2	1508.60	1508.597	0.00
A_1	ν_2	1746.1	1749.66	1749.662	0.00
A_1	$2\nu_4$	2327.5	2332.71	2332.716	-0.01
A_2	$\nu_4 + \nu_6$	2422.4	2431.05	2431.051	0.00
A_1	$2\nu_6$	2496.1	2501.93	2501.928	0.00
B_1	$\nu_3 + \nu_4$	2667.1	2679.48	2679.477	0.00
B_2	$\nu_3 + \nu_6$	2718.6	2728.98	2728.978	0.00
A_1	ν_1	2782.2	2782.84	2782.834	0.01
B_2	ν_5	2843.0	2841.67	2841.661	0.01
B_1	$\nu_2 + \nu_4$	2906.0	2913.26	2913.259	0.00

Results

sym	Assig	Exp	Luckhaus (b)	This work (c)	$b-c$
A_1	$2\nu_3$	2998.1	3015.72	3015.718	0.00
B_2	$\nu_2+\nu_6$	3000.6	3006.73	3006.724	0.01
A_1	$\nu_2+\nu_3$	3239.0	3250.41	3250.402	0.01
A_1	$2\nu_2$	3471.6	3479.87	3479.865	0.01
B_1	$3\nu_4$	3480.7	3485.00	3485.003	0.00
B_2	$2\nu_4+\nu_6$	3586.6	3596.66	3596.665	-0.01
B_1	$\nu_4+2\nu_6$	3673.5	3688.91	3688.910	0.00
B_2	$3\nu_6$		3747.46	3747.460	0.00
A_1	$\nu_3+2\nu_4$	3825.3	3840.19	3840.192	0.00
A_2	$\nu_3+\nu_4+\nu_6$	3886.5	3902.61	3902.604	0.01
A_1	$\nu_3+2\nu_6$	3937.4	3948.77	3948.759	0.01
B_1	$\nu_1+\nu_4$	3940.2	3946.77	3946.763	0.01

Results

sym	Assig	Exp	Luckhaus (b)	This work (c)	$b-c$
A_2	$\nu_4+\nu_5$	3995.8	4001.17	4001.157	0.01
B_2	$\nu_1+\nu_6$		4027.23	4027.211	0.02
A_1	$\nu_2+2\nu_4$	4058.3	4066.64	4066.641	0.00
A_1	$\nu_5+\nu_6$	4083.1	4088.52	4088.499	0.02
A_2	$\nu_2+\nu_4$ $+\nu_6$	4163.9	4175.55	4175.540	0.01
B_1	$2\nu_3+\nu_4$		4186.18	4186.178	0.00
B_2	$2\nu_3+\nu_6$		4210.67	4210.645	0.03
A_1	$\nu_2+2\nu_6$	4248.7	4257.93	4257.923	0.01
A_1	$\nu_1+\nu_3$	4253.8	4265.94	4265.906	0.03
B_2	$\nu_3+\nu_5$	4335.1	4345.10	4345.066	0.03
B_1	$\nu_2+\nu_3$ $+\nu_4$	4397.5	4413.60	4413.591	0.01

Results

sym	Assig	Exp	Luckhaus (b)	This work (c)	$b-c$
B_2	$\nu_2+\nu_3$ $+\nu_6$	4466.8	4479.98	4479.949	0.03
A_1	$3\nu_3$		4520.35	4520.328	0.02
A_1	$\nu_1+\nu_2$	4529.4	4533.11	4533.063	0.05
B_2	$\nu_2+\nu_5$	4571.5	4573.60	4573.555	0.05
B_1	$2\nu_2+\nu_4$	4624.3	4635.90	4635.883	0.02
A_1	$4\nu_4$	4629.0	4627.89	4627.888	0.00
A_1	$\nu_2+2\nu_3$	4730.8	4749.30	4749.263	0.04
B_2	$2\nu_2+\nu_6$	4733.8	4744.14	4744.108	0.03
A_2	$3\nu_4+\nu_6$	4741.9	4751.76	4751.761	0.00
A_1	$2\nu_4+2\nu_6$	4842.0	4858.07	4858.069	0.00
A_2	$\nu_4+3\nu_6$		4945.24	4945.238	0.00
A_1	$2\nu_2+\nu_3$	4955.2	4972.31	4972.265	0.05
B_1	$\nu_3+3\nu_4$	4977.1	4991.38	4991.380	0.00

Results

sym	Assig	Exp	Luckhaus (b)	This work (c)	$b - c$
A_1	$4\nu_6$		4990.19	4990.178	0.01
B_2	$\nu_3 + 2\nu_4 + \nu_6$	5043.7	5060.42	5060.409	0.01
A_1	$\nu_1 + 2\nu_4$	5092.4	5100.25	5100.243	0.01
B_1	$\nu_3 + \nu_4 + 2\nu_6$	5104.0	5129.87	5129.855	0.01
B_2	$2\nu_4 + \nu_5$	5140.1	5150.73	5150.710	0.02
B_2	$\nu_3 + 3\nu_6$	5151.0	5168.21	5168.174	0.04
A_1	$3\nu_2$	5177.6	5191.42	5191.382	0.04
A_2	$\nu_1 + \nu_4 + \nu_6$		5197.60	5197.575	0.02
B_1	$\nu_2 + 3\nu_4$	5205.2	5211.64	5211.638	0.00

Results

sym	Assig	Exp	Luckhaus (b)	This work (c)	$b-c$
B_1	$\nu_4+\nu_5+\nu_6$	5244.1	5258.99	5258.969	0.02
A_1	$\nu_1+2\nu_6$		5265.19	5265.134	0.06
B_2	$\nu_5+2\nu_6$	5312.2	5324.82	5324.788	0.03
A_1	$2\nu_3+2\nu_4$	5321.3	5345.49	5345.469	0.02
B_2	$\nu_2+2\nu_4+\nu_6$	5325.6	5336.91	5336.893	0.02
A_2	$2\nu_3+\nu_4+\nu_6$	5353.2	5382.63	5382.600	0.03
A_1	$2\nu_3+2\nu_6$	5389.4	5404.04	5403.954	0.09
B_1	$\nu_1+\nu_3+\nu_4$		5430.42	5430.388	0.03
B_1	$\nu_2+\nu_4+2\nu_6$	5417.6	5434.25	5434.234	0.02

Results

sym	Assig	Exp	Luckhaus (b)	This work (c)	$b - c$
B_2	$\nu_1 + \nu_3$				
	$+\nu_6$	5433.4	5442.37	5442.232	0.14
A_1	$2\nu_1$	5462.7	5472.06	5471.900	0.16
B_2	$\nu_2 + 3\nu_6$		5504.12	5504.070	0.05
A_2	$\nu_3 + \nu_4$				
	$+\nu_5$	5489.0	5506.75	5506.717	0.03
B_2	$\nu_1 + \nu_5$	5530.5	5542.45	5542.290	0.16
A_1	$\nu_2 + \nu_3$				
	$+2\nu_4$	5546.5	5567.85	5567.824	0.03
A_1	$\nu_3 + \nu_5$				
	$+\nu_6$	5551.3	5557.47	5557.357	0.11
A_2	$\nu_2 + \nu_3$				
	$+\nu_4 + \nu_6$	5625.5	5644.84	5644.797	0.04
A_1	$2\nu_5$	5651.0	5653.58	5653.395	0.18

If the potential is a SOP it is possible to drastically reduce the memory cost by using different ideas

$$\boxed{n^{2D} \rightarrow n^{D} \rightarrow \frac{(D+n)!}{D!n!} \rightarrow n}$$

If $D = 20$ and $n = 15$

$$\boxed{9 \times 10^{38} GB \rightarrow 3 \times 10^{15} GB \rightarrow 26 GB \rightarrow 1 \text{ GB}}$$

Use SOP basis functions

A basis function,

$$\Psi(q_1, \ldots, q_D) = \sum_{i_1=1}^{n_1} \cdots \sum_{i_D=1}^{n_D} F_{i_1 i_2 \ldots i_D} \prod_{j=1}^{D} \theta_{i_j}^{j}(q_j) \ .$$

is itself a SOP if

$$F_{i_1 i_2 \ldots i_D} = \sum_{\ell=1}^{R} \prod_{j=1}^{D} f_{i_j}^{(\ell,j)} \ ,$$

The memory cost scales as RnD

The canonical polyadic (CP) decomposition for tensors.

How do we make the SOP basis functions?

- Start with $F_{i_1 i_2 \ldots i_D} = \prod_{j=1}^{D} f_{i_j}^{(1,j)}$,

  with some random $f_{i_j}^{(1,j)}$

- the Hamiltonian is

$$\hat{H}(q_1, \ldots, q_D) = \sum_{k=1}^{T} \prod_{j=1}^{D} h_{kj}(q_j),$$

- Make basis vectors by applying **H** to $F_{i_1 i_2 \ldots i_D}$

Block Power Method

Make a basis by applying $\widetilde{\mathbf{H}}$ to a set of B start vectors.

$$\widetilde{H} = H - \sigma I$$

Alternating successive applications of $\widetilde{\mathbf{H}}$ with a modified Gram-Schmidt orthogonalization, we obtains a basis of SOP vectors,

Block Power Method

The key step is

$$(\mathbf{F}')_{i'_1 i'_2 \cdots i'_D} = (\mathbf{HF})_{i'_1 \dots i'_D}$$

$$\begin{aligned}
&(\mathbf{F}')_{i'_1 i'_2 \cdots i'_D} = (\mathbf{HF})_{i'_1 \dots i'_D} \\
&= \sum_{i_1, i_2, \cdots, i_D} \sum_{k=1}^{T} \prod_{j'=1}^{D} (\mathbf{h}_{kj'})_{i'_{j'} i_{j'}} \sum_{\ell=1}^{R} \prod_{j=1}^{D} f_{i_j}^{(\ell, j)} \\
&= \sum_{k=1}^{T} \sum_{\ell=1}^{R} \prod_{j=1}^{D} \sum_{i_j} (\mathbf{h}_{kj})_{i'_j i_j} f_{i_j}^{(\ell, j)}.
\end{aligned}$$

Only 1-D matrix-vector products are required

The rank increases

The rank of $\mathbf{F}'$ is a factor of T larger than the rank of $\mathbf{F}$.

Applying $\mathbf{H}$ to $\mathbf{F}$, with R terms, yields a vector with RT terms.

The rank must therefore be reduced after each matrix-vector product.

Rank reduction

To reduce the rank, we use an alternating least squares algorithm of Beylkin and Mohlenkamp to replace

$$\begin{aligned} F^{\text{old}}_{i_1 i_2 \ldots i_D} &= \sum_{\ell=1}^{R_{\text{old}}} \prod_{j=1}^{D} {}^{\text{old}}f^{(\ell,j)}_{i_j} \\ &\implies F^{\text{new}}_{i_1 i_2 \ldots i_D} = \sum_{\ell=1}^{R_{\text{new}}} \prod_{j=1}^{D} {}^{\text{new}}f^{(\ell,j)}_{i_j} \ , \end{aligned}$$

by choosing $f^{(\ell,j)}_{i_j}$ to minimize $\| \mathbf{F}^{\text{new}} - \mathbf{F}^{\text{old}} \|$.

Results

$$H(q_1, \ldots, q_D) = \sum_{j=1}^{D} \frac{\omega_j}{2} \left(p_j^2 + q_j^2\right) + \sum_{\substack{i,j=1 \\ i>j}}^{D} \alpha_{ij} q_i q_j$$

$$\omega_j = \sqrt{j/2},\ j = 1, \ldots, 6.$$

$\alpha_{ij} = 0.1$
$D = 20$
$n = 10$
Reduction rank $= 20$
Block size $= 56$
Max$(N_{pow}) = 5000$

Memory cost

A single vector has 10^{20} components ; 8×10^{11} GB.

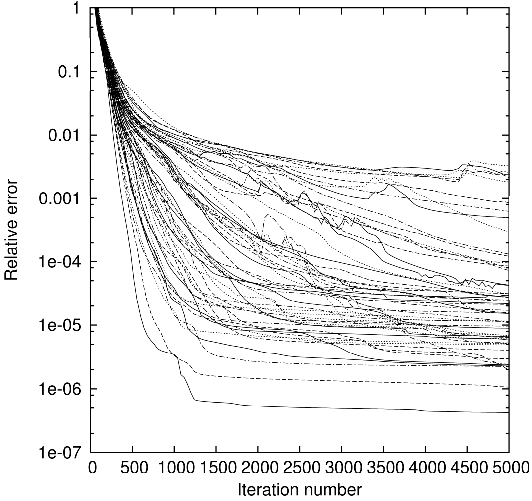
With the rank reduction method, we require less than 1 GB

Table: Energy levels of the 20 coupled oscillator Hamiltonian. From left to right : energy level number, exact energy level, RRBPM energy level, relative error, normal mode assignment.

n	$E_{n,\mathrm{th}}$	$E_{n,\mathrm{num}}$	$\frac{E_{n,\mathrm{num}}-E_{n,\mathrm{th}}}{E_{n,\mathrm{th}}}$	Assignment
0	21.719578	21.719587	4.2×10^{-7}	-
1	22.398270	22.398294	1.1×10^{-6}	ν_1
2	22.691775	22.691826	2.2×10^{-6}	ν_2
3	22.917012	22.917129	5.1×10^{-6}	ν_3
4	23.076962	23.077014	2.3×10^{-6}	$2\,\nu_1$
5	23.106960	23.107006	2.0×10^{-6}	ν_4
6	23.274380	23.274502	5.3×10^{-6}	ν_5
7	23.370467	23.370629	6.9×10^{-6}	$\nu_1+\nu_2$
8	23.425814	23.425951	5.8×10^{-6}	ν_6
9	23.565153	23.565222	2.9×10^{-6}	ν_7
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

n	$E_{n,\mathrm{th}}$	$E_{n,\mathrm{num}}$	$\frac{E_{n,\mathrm{num}}-E_{n,\mathrm{th}}}{E_{n,\mathrm{th}}}$	Assignment
20	24.049160	24.049374	8.9×10^{-6}	$2\,\nu_1+\nu_2$
21	24.079158	24.079914	3.1×10^{-5}	$\nu_3+\nu_4$
22	24.104506	24.104878	1.6×10^{-5}	$\nu_1+\nu_6$
23	24.114446	24.114570	5.2×10^{-6}	$2\,\nu_3$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
30	24.342665	24.343080	1.7×10^{-5}	$\nu_1+2\,\nu_2$
31	24.346217	24.346365	6.1×10^{-6}	ν_{14}
32	24.373625	24.373996	1.5×10^{-5}	$\nu_1+\nu_8$
33	24.398012	24.398676	2.7×10^{-5}	$\nu_2+\nu_6$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

n	$E_{n,\mathrm{th}}$	$E_{n,\mathrm{num}}$	$\frac{E_{n,\mathrm{num}}-E_{n,\mathrm{th}}}{E_{n,\mathrm{th}}}$	Assignment
40	24.532333	24.533376	4.3×10^{-5}	ν_{16}
41	24.537351	24.539130	7.3×10^{-5}	$\nu_2+\nu_7$
42	24.567902	24.570246	9.5×10^{-5}	$\nu_1+\nu_2+\nu_3$
43	24.611100	24.613013	7.8×10^{-5}	$\nu_1+\nu_{10}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
50	24.709939	24.725314	6.2×10^{-4}	ν_{18}
51	24.721068	24.765667	1.8×10^{-3}	$\nu_1+\nu_{11}$
52	24.727852	24.786084	2.4×10^{-3}	$3\,\nu_1+\nu_2$
53	24.757850	24.810515	2.1×10^{-3}	$\nu_1+\nu_2+\nu_4$
54	24.762587	24.823982	2.5×10^{-3}	$\nu_3+\nu_7$
55	24.783198	24.863299	3.2×10^{-3}	$2\,\nu_1+\nu_6$


Relative error
1
0.1
0.01
0.001
1e-04
1e-05
1e-06
1e-07
0
500
1000
1500
2000
2500
3000
3500
4000
4500
5000
Iteration number

Conclusion

- Smolyak ideas, whether used to do quadrature or with collocation, significantly reduce the memory cost of computing vibrational spectra
- Polynomial basis functions are much better than piecewise linear functions
- With collocation one can use non-orthogonal basis functions and obviate the need to solve a generalized eigenvalue problem.
- 12-D calculations are possible.
  Grid size $\sim 5.7 \times 10^{13} \rightarrow 8.5 \times 10^{6}$
  Memory cost 500 GB $\rightarrow$ 0.07 GB
- When the potential is simple, SOP basis functions, a shifted power method, and rank reduction scheme make 20-D calculations are possible with a memory cost of less than 1 GB.

This work has been supported by the Natural Sciences and Engineering Research Council of Canada, the Réseau québécois de calcul de haute performance, and the Canada Research Chairs programme.