Coping with the curse of dimensionality when computing a vibrational spectrum for a general potential

Gustavo Avila and Tucker Carrington

Chemistry Department
Queen's University

3 juin 2015

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

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

Warning : I am solving the VIBRATIONAL Schroedinger equation

General potentials

In general, the potential, $\hat{V}$, is a complicated function. When it cannot be represented as

- a sum-of-products
- a sum of terms with one, two, etc coordinates

we have used two methods :

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

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

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,

$$H = H_0 + \Delta$$

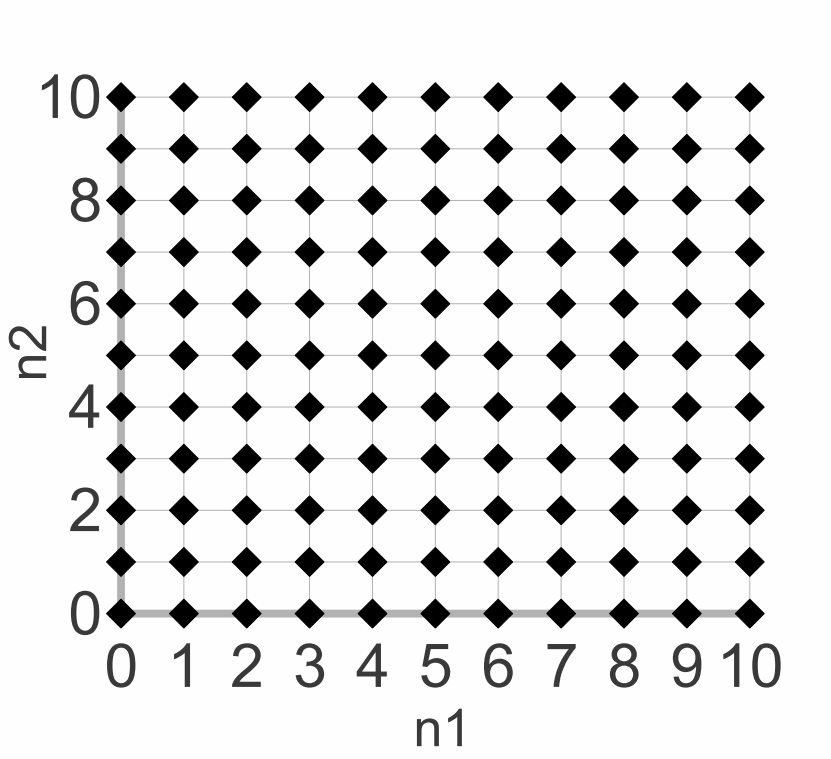
H_0 is a sum of 1d Hamiltonians (separable).

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

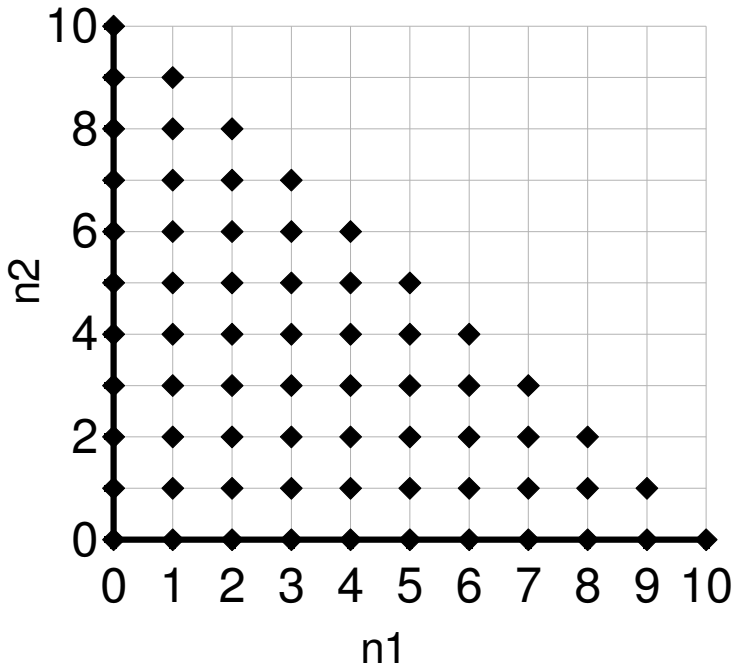
If all the 1d Hamiltonians are identical one simply removes basis functions for which

$$\sum_c n_c > b$$

Full basis for a 2d problem



Pruned basis for the 2d problem



Better than the hyperbolic cross

- If I want the lowest eigenvalues of H and if Δ is small then the $\sum_c n_c \leq b$ basis must be better
- Some of the eigenfunctions I desire have nodes in many (all) coordinates

One can do better

The simplest basis pruning condition is

$$n_1 + \cdots + n_D \leq b.$$

We have also used the pruning condition,

$$\alpha_1 n_1 + \cdots + \alpha_D n_D \leq b.$$

We have also used more general pruning conditions of the form

$$g^1(n_1) + \cdots + g^D(n_D) \leq b \ ,$$

where $g^c(n)$ is an arbitrary monotonic function of n, designed to include basis functions coupled by the largest terms in the Hamiltonian.

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.

Basis vector : 3×10^9 GB $\rightarrow$ 0.6 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 $f(x_1, x_2, \cdots, x_D)$ can be written as a sum of D-dimensional product quadrature grids,

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

Crucial implementation details

- We use nested grids and obviate the need to sum over contributing grids,

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

- With nested points a Smolyak quadrature can be written,

$$\begin{aligned} & S(6,H) f(q_1, q_2, q_3, q_4, q_5, q_6) \\ = & \sum_{k_1}^{N_1} \sum_{k_2}^{N_2} \sum_{k_3}^{N_3} \sum_{k_4}^{N_4} \sum_{k_5}^{N_5} \sum_{k_6}^{N_6} w(k_6, k_5, k_4, k_3, k_2, k_1) \\ & \times f(q_1^{k_1}, q_2^{k_2}, q_3^{k_3}, q_4^{k_4}, q_5^{k_5}, q_6^{k_6}) \end{aligned} ,$$

where

$$w(k_6, \cdots, k_1) = \sum_{f(\mathbf{i}) \leq H} C_{i_1,\cdots,i_6}{}^{i_1} w_{k_1} \cdots {}^{i_D} w_{k_6},$$

We exploit the structure of the basis and the grid to evaluate matrix-vector products by doing sums sequentially

$$v2(n_3', n_2', n_1') = \sum_{k_1=1}^{N_1} T_{n_1' k_1} \sum_{k_2=1}^{N_2} T_{n_2' k_2} \sum_{k_3=1}^{N_3} T_{n_3' k_3}$$
$$w(k_3, k_2, k_1) V(q_1^{k_1}, q_2^{k_2}, q_3^{k_3})$$
$$\sum_{n_3=0}^{n_3^{\max}} T_{n_3 k_3} \sum_{n_2=0}^{n_2^{\max}} T_{n_2 k_2} \sum_{n_1=0}^{n_1^{\max}} T_{n_1 k_1}$$
$$v1(n_3, n_2, n_1) \ ,$$

where $T_{nk} = \phi_n(q_k)$

Crucial advantage that there is no sum over contributing grids

Smolyak quadratures adapted to our bases work well for a 12-D problem

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

memory cost 500 TB $\rightarrow$ 0.07 GB

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 $\mathbf{HU} = \mathbf{SUE}$

We have also used collocation

Collocation obviates

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

Collocation has the advantage that as the basis improves, the choice of the points becomes irrelevant.

What is collocation ?

Apply $\hat{H} - E_k$ to

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

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

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

Two drawbacks of established collocation methods

- $\mathbf{B} \neq \mathbf{I}$
  and it is necessary to solve a generalized eigenvalue problem

- The $\mathbf{H}$ and $\mathbf{B}$ 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 span the same space as the first m harmonic oscillator functions,

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

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 $M_{k'q} = \phi_q(x_{k'})$.

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.

$$I(D,H)\Phi_n(x_1,x_2,\cdots,x_D) = \sum_{g(i_1,i_2,\cdots,i_D)\leq H} C_{i_1,i_2,\cdots,i_D}$$
$$\times U^{i_1}(x_1)\otimes U^{i_2}(x_2)\otimes\cdots U^{i_D}(x_c)\Phi_n(x_{k_1},x_{k_2},\cdots,x_{k_D})$$

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

$C_{i_1,i_2,\cdots,i_D}$ coefficients are the same as those used with Smolyak quadrature.

Multidimensional collocation

The interpolant is built using multiple sets of nested points,

$$
\begin{aligned}
x_1^c \text{ for } i_c &= 1, \\
x_1^c, x_2^c \text{ for } i_c &= 2, \\
x_1^c, x_2^c, x_3^c \text{ for } i_c &= 3, \\
x_1^c, x_2^c, x_3^c, x_4^c \text{ for } i_c &= 4, \\
&\vdots \\
x_1^c, x_2^c, \cdots, x_{K-1}^c, x_K^c \text{ for } i_c &= K \ .
\end{aligned}
$$

Multidimensional collocation

Again, one can use

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

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.

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

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

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

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

We convert this into a matrix eigenvalue problem

- $$(\mathbf{T} + \mathbf{V})\mathbf{U} = \mathbf{U}\mathbf{E},$$

- We do not construct $\mathbf{T}$; $\mathbf{V}$ is diagonal.
- Elements of the eigenvectors are values of wavefunctions at points.
- As H is increased, diagonal elements of $\mathbf{E}$ and columns of $\mathbf{U}$ 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.

Kinetic energy matrix-vector products

$$K = \sum_{k,l}^{3N-6} \left(G_{kl} \frac{\partial}{\partial Q_k} \frac{\partial}{\partial Q_l} \right) + \sum_{k}^{3N-6} \left(H_k \frac{\partial}{\partial Q_k} \right) + V'$$

- Apply the KEO term by term
- Cost of computing the spectrum does not depend on the form or complexity of the functions G_{kl} and H_k

Avoid $C_{i_1,i_2,\cdots,i_D}$

It is important not to use

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

- sum over contributing grids
- $C_{i_1,i_2,\cdots,i_D}$ makes efficient sequential summation impossible

Rather than writing an interpolated wavefunction as a sum over points,

$$I(D,H)\Phi_n(x_1,x_2,\cdots,x_D) = \sum_{g(i_1,i_2,\cdots,i_D)\le H} C_{i_1,i_2,\cdots,i_D} \times U^{i_1}(x_1)\otimes U^{i_2}(x_2)\otimes\cdots U^{i_D}(x_c)\Phi_n(x_{k_1},x_{k_2},\cdots,x_{k_D})$$

where U^{i_c} is a 1-D interpolant defined by

$$f(x_c)\approx U^{i_c}f(x_c)=\sum_{k_c=1}^{m_{i_c}} f(x_{k_c})a_{k_c}^{i_c}(x_c).$$

it can be written as

$$I(D,H)\Phi_n(x_1,x_2,\cdots,x_D) = \sum_{g(n_1+1,n_2+1,\cdots,n_D+1)\le H} {}^nC^{or}_{n_1,\cdots,n_D}\ \varphi_{n_1}(x_1)\varphi_{n_2}(x_2)\cdots\varphi_{n_D}(x_D)$$

- When the interpolated wavefunction is written as a linear combination of products of $\varphi_{n_c}(x_c)$, the KEO can be applied by differentiating $\varphi_{n_c}(x_c)$.
- This is more efficient than applying the KEO to the point form of the interpolated wavefunction.
- For chemists : it is easier to apply the KEO to an FBR representation of the wavefunction than to a DVR representation of the wavefunction.
- To use this idea one must obtain ${}^{n}C^{or}_{n_1,\cdots,n_D}$ from $\Phi_n(x_{k_1}, x_{k_2}, \cdots, x_{k_D})$
- This is not simple because neither the basis nor the grid is a direct product (both are direct products in a standard pseudo-spectral calculation).

How to do the transformation

We use a recursive formulation

(everything here for the simple case
$g(i_1, i_2, \cdots, i_D) = i_1 + i_2 + \cdots + i_D \leq H$)

$$I(D, H) = \sum_{p=1}^{H-D+1} I(D-1, H-p)(U^p - U^{p-1}).$$

For example, in six dimensions, the 6D Smolyak interpolant is

$$I(6, H) = \sum_{i_6=1}^{H-5} I(5, H-i_6)[U^{i_6}(x_6) - U^{i_6-1}(x_6)] \ .$$

Writing the $a_{k_c}^{i_c}$ in terms of $\varphi_{n_c}(x_c)$

$$a_{k_c}^{i_c}(x_c) = \sum_{n_c=0}^{n_c^{max}(i_c)} B_{k_c,n_c}^{i_c} \varphi_{n_c}(x_c),$$

where

$$B^{i_c} = (P^{i_c})^{-1}$$

$$P_{n_c,k_c}^{i_c} = \varphi_{n_c}(x_c^{k_c})$$

- $\Delta B^{i_c}_{k_c,n_c}$ factors are applied sequentially for all the coordinates
- Rather than using $\varphi_{n_c}(x_c)$, it is better to use $\tilde{\varphi}_{n_c}(x_c)$ that span, level by level, the same space(s) and defined so that $\varphi_{n_c}(x_c^{k_c})$ =0 when $k_c < n_c - 1$

$$\tilde{\varphi}_{n_c}(x_c) = \sum_{m_c=1}^{n_c^{max}-1} \tilde{A}_{n_c,m_c}\varphi_{m_c}(x_c) + \varphi_{n_c}(x_c),$$

- many $\Delta\tilde{B}^{i_c}_{k_c,n_c}$ are zero
- Sums are evaluated sequentially, exploiting the structure of the pruned basis and the Smolyak grid. The cost scales as n^{D+1}.

Test calculations

- CH_2O with normal coordinates
- HONO with bond length and bond angles coordinates
- Cut eigenfunctions for 1-D bases

CH_2O

For CH_2O, the 100 lowest levels on the PES of Martin et al. are accurately calculated with

$H = 25$ and 177'100 points.

For HONO, the 330 lowest levels on the PES of Richter et al. are accurately calculated with

$H = 27$ and 2×10^6 points.

HONO is a harder problem because it is a double-well potential.

Errors $\sim 0.3 \quad \text{cm}^{-1}$ (one part in 10^4)

Points of possible interest to mathematicians

- General pruning of the basis and the grid,

$$g^1(n_1) + \cdots + g^D(n_D) \leq b \ ,$$

- In the Galerkin case, obviate the sum over grids by using Smolyak weights

$$w(k_6, \cdots, k_1) = \sum_{f(\mathbf{i}) \leq H} C_{i_1,\cdots,i_6}{}^{i_1} w_{k_1} \cdots {}^{i_D} w_{k_6},$$

Points of possible interest to mathematicians

- Rather than transforming to the "nodal" basis and using the "unidirectional " principle, we do sums sequentially. There is no need for the operator to be a sum of products

$$v2(n_3', n_2', n_1') = \sum_{k_1=1}^{N_1} T_{n_1' k_1} \sum_{k_2=1}^{N_2} T_{n_2' k_2} \sum_{k_3=1}^{N_3} T_{n_3' k_3}$$
$$w(k_3, k_2, k_1) V(q_1^{k_1}, q_2^{k_2}, q_3^{k_3})$$
$$\sum_{n_3=0}^{n_3^{\max}} T_{n_3 k_3} \sum_{n_2=0}^{n_2^{\max}} T_{n_2 k_2} \sum_{n_1=0}^{n_1^{\max}} T_{n_1 k_1}$$
$$v1(n_3, n_2, n_1) \ ,$$

- No hash tables, no trees ; we have an equation for $m(n_1, n_2, \cdots)$; for the potential we just use counters

Points of possible interest to mathematicians

- We use nested points, but abhor 2^{l+1} ; the difference between the number of points in level $l+1$ and level l is very small (often one)
- We use various point sets, but often choose points to optimize the calculation of the (1-d) overlap (Gram) matrix

Points of possible interest to mathematicians

- Multidimensional collocation without a “mass matrix”, crucial if iterative eigensolvers are necessary
- Collocation with spectral Lagrange-type functions

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.