0.1 Laplace Denominators

0.1.1 Approximation of $1/x$ on $[0, \infty)$ by Exponential Sums

A very useful identity is,

$$\frac{1}{x} = \int_0^\infty \mathrm{d}\alpha \; e^{-x\alpha} \approx \sum_w \omega_w e^{-\alpha_w x}$$

The last line indicates that a quadrature rule $\{< \alpha_w, \omega_w >\}$ is being used to approximate the integral over α. This understanding of the Laplace resolution is due to Almlof,[1] who explored a variety of quadrature rules to approximate the integral.

More recently, Braess and Hackbusch have realized that it is better to directly fit $1/x$ as a sum of exponentials for x in the range of $[1, R)$,[2] rather than using the quadrature interpretation,

$$\frac{1}{x} \approx \sum_w \omega_w e^{-\alpha_w x} : \left| \frac{1}{x} - \sum_w \omega_w e^{-\alpha_w x} \right| < \delta \Bigg|_{x \in [1,R)}$$

He has located the optimal quadrature rule by the Remez minimax algorithm for a variety of R and for n_w between 1 and 50, and has published the quadrature rules $\{< \alpha_w, \omega_w >\}$ and corresponding maximum error δ for each case.[3]

0.1.2 Laplace Denominators in Electronic Structure Theory

We start from a set of "occupied" orbitals labeled by i, j, k, l with orbital energies $\{\epsilon_i\}$ and a set of "virtual" orbitals labeled by a, b, c, d with orbital energies $\{\epsilon_a\}$. We assert that the occupied orbital energies are strictly less than the virtual orbital energies, i.e., $\epsilon_i < \epsilon_a \; \forall \; i, a$.

A second-order orbital energy denominator has the form,

$$\Delta_{ijab} \equiv \frac{1}{\epsilon_a + \epsilon_b - \epsilon_i - \epsilon_j} \approx \bar{\tau}_i^w \bar{\tau}_j^w \tau_a^w \tau_b^w$$

A third-order orbital energy denominator has the form,

$$\Delta_{ijkabc} \equiv \frac{1}{\epsilon_a + \epsilon_b + \epsilon_c - \epsilon_i - \epsilon_j - \epsilon_k} \approx \bar{\tau}_i^w \bar{\tau}_j^w \bar{\tau}_k^w \tau_a^w \tau_b^w \tau_c^w$$

In an n-th order orbital energy denominator, we define the "extent" of the argument x to be,

$$R \equiv \frac{\max(\epsilon_a) - \min(\epsilon_i)}{\min(\epsilon_a) - \max(\epsilon_i)}$$

[1] J. Almlöf, *Chem. Phys. Lett.*, **181**, 319 (1991)
[2] D. Braess and W. Hackbusch, *IMA J. Numer. Anal.*, **25**, 685 (2005)
[3] See Hackbusch's website.

Using this and a user-specified error criterion δ, we select one of the Hackbush quadratures $\{< \alpha_w, \omega_w >\}$ for use on $[1, R)$. We now define a scaling constant,[4]

$$\gamma \equiv n(\min(\epsilon_a) - \max(\epsilon_i))$$

We now define the modified quadrature rule,

$$\alpha'_w \equiv \alpha_w/\gamma, \ \omega'_w \equiv \omega_w/\gamma$$

And arrive at, e.g.,

$$\Delta_{ijab} \approx \sum_w \omega'_w \exp(-\alpha'_w(\epsilon_a + \epsilon_b - \epsilon_i - \epsilon_j))$$

Finally, by inspection,

$$\bar{\tau}_i^w \equiv \sqrt[2n]{\omega'_w} e^{+\alpha'_w \epsilon_i}$$

$$\tau_a^w \equiv \sqrt[2n]{\omega'_w} e^{-\alpha'_w \epsilon_a}$$

Note that whereas Hackbush's quadrature rules bound the maximum *absolute* error of $1/x$ to δ for $x \in [1, R)$, our choice of quadrature selection and scaling bounds the maximum *relative* error of Δ_{ijab} to δ for the given orbital eigenvalues.

1 Density Fitting

A key quantity in electronic structure theory is the electron repulsion integral (ERI),

$$(pq|rs) \equiv \iint_{\mathbb{R}^6} \mathrm{d}^3 r_1 \ \mathrm{d}^3 r_2 \ \phi_p(\vec{r}_1)\phi_q(\vec{r}_1)\frac{1}{r_{12}}\phi_r(\vec{r}_2)\phi_s(\vec{r}_2)$$

Here $\{\phi_p(\vec{r}_1)\}$ is the atomic-orbital basis of atom-centered Gaussian functions (contracted Cartesian or real solid harmonic basis functions). Analytical formulae exist for the ERIs, but they are a major bottleneck to generate, store, and manipulate.

In density fitting,[5] we approximate the electron repulsion integrals

$$(pq|rs) \approx (pq|A)(A|B)^{-1}(B|rs) = (pq|A)(A|C)^{-1/2}(C|B)^{-1/2}(B|rs) \equiv L_{pq}^C L_{rs}^C$$

Here $\{\chi_A(\vec{r}_1)\}$ is an auxiliary basis set of atom-centered Gaussian functions. This is usually defined to accompany a given primary AO basis set, and has typically $2-5\times$ the number of basis functions as the primary basis.

[4] What we are doing here is to write

$$\frac{1}{\eta} = \gamma \frac{1}{\eta/\gamma}$$

with γ chosen so that the maximum argument of the second denominator is 1. This simple rescaling then lets us use Hackbusch's quadrature rules.

[5] Specifically robust Coulomb-metric density fitting.

In the above, the two-index integrals (often called the DF metric) are,

$$(A|B) \equiv \iint_{\mathbb{R}^6} \mathrm{d}^3 r_1 \ \mathrm{d}^3 r_2 \ \chi_A(\vec{r}_1) \frac{1}{r_{12}} \chi_B(\vec{r}_2)$$

and the three-index integrals are,

$$(A|pq) \equiv \iint_{\mathbb{R}^6} \mathrm{d}^3 r_1 \ \mathrm{d}^3 r_2 \ \chi_A(\vec{r}_1) \frac{1}{r_{12}} \phi_p(\vec{r}_1) \phi_q(\vec{r}_2)$$

If one has formulae (or codes) for the four-center integrals $(pq|rs)$, one can very easily obtain the two- and three-center integrals by substituting "null" basis functions $\varnothing(\vec{r}_1) \equiv 1$ (a single Gaussian s function at the origin with exponent of 0 and prefactor of 1) into the expressions for the four-center integrals, e.g., $(A|B) \equiv (A\varnothing|B\varnothing)$ or $(A|pq) \equiv (A\varnothing|pq)$. The Cauchy-Schwarz bound can be exploited to only form (and, optionally, only store) significant pq pairs in the three-index integrals above.[6]

The form seen above in terms of L^C_{pq} is the "symmetric" form of the DF approximation, and involves applying the inverse square root of the metric to the three-index integrals. In practice, the auxiliary basis may be rather ill-conditioned in the Coulomb metric, but this is easily dealt with by discarding the smallest eigenvectors in the metric to preserve an inverse relative condition number of κ in the basis,

$$(A|B)^{-1/2} \equiv U_{AC} s_C^{-1/2} U_{BC} : \ s_C > \kappa s_C^{\max}, \ (A|B) = U_{AC} s_C U_{BC}$$

Here U_{AC} are the eigenvectors and S_C are the eigenvalues of the metric $(A|B)$. The value of κ is specified by the user, and should be small enough that significant parts of the auxiliary basis are not discarded, but large enough that numerical precision artifacts do not affect the result. $\kappa = 10^{-12}$ seems to be a reasonable choice across a vast range of EST problems, provided that double precision is used.

1.1 AO Density Fitting

Our objective in `ao_df` is to produce the tensor,

$$L^C_{pq} \equiv (C|A)^{-1/2} (A|pq)$$

The tensor should be stored as Apq, and will not exploit spatial sparsity or permutational symmetry.

1. Form the metric matrix $(C|A)$.
2. Form the metric inverse square root $(C|A)^{-1/2}$ via the `Tensor::power` method. Be sure to condition the inverse via the user-supplied inverse relative condition number κ.

[6] The LS `PairList` object tells you about the significant shell pairs. We throw away whole shell pairs to preserve rotational invariance.

3. Allocate a single tensor `L` with indices Apq.
4. Form the integrals $(A|pq)$ and place into `L`. Do not compute insignificant integrals
5. Perform the matrix multiplication $(C|A)^{-1/2}(A|pq)$ in chunks of pq (using chunks of N_A pq indices works great). Copy the chunk from `L` into a temporary buffer, and then `dgemm` it back into `L`.

Overall, this uses a single copy of size Apq, plus two copies of $(C|A)$. And it is fairly simple as such algorithms go. E.g., we could have avoided a memcpy in the last stage by directly computing the integrals in the `dgemm` buffer. But then we would have had issues with shell granularity of the integrals.

1.2 MO Density Fitting

Our objective in `mo_df` is to produce the tensor,

$$L_{ia}^{C} \equiv (C|A)^{-1/2}(A|pq)C_{pi}\bar{C}_{pa}$$

The tensor should be stored as iaC. A number of combinations of C_{pi} and $\bar{C}_{qa}$ should be allowed. The transformation in C_{pi} should be performed first, and the documentation should recommend using the smaller (usually occupied) index for the first part of the transformation, to save FLOPS/storage.

1. Form the metric matrix $(C|A)$.
2. Form the metric inverse square root $(C|A)^{-1/2}$ via the `Tensor::power` method. Be sure to condition the inverse via the user-supplied inverse relative condition number κ.
3. Allocate a temporary copy of the target tensors with indices Aia
4. For stripes of A (up to say 2 GB, using whole A shells):
   (a) Form the integrals $(A|pq)$ for the stripe. Do not compute insignificant integrals.
   (b) Perform the contraction $(A|pi) = (A|pq)C_{qi}$ with one `dgemm` call for the stripe.
   (c) Perform the contraction $(A|ia) = (A|pi)\bar{C}_{pa}$ with N_A^{stripe} `dgemm` calls. The result should be placed directly in the temporary target tensors.
5. Perform the fitting $L_{ia}^{C} = (C|A)^{-1/2}(A|ia)$ for each target tensor. The striping will reverse from Aia to iaC in this step. An extra copy of size iaC can be used in this step.

Here the integrals $(A|pq)$ can be used for all combinations of C_{pi} and $\bar{C}_{qa}$. If one is clever, the first-half transform in 4b can be re-used for tasks with the same C_{pi} tensor.