SCALING DENSITY FUNCTIONAL THEORY WITH GAUSSIAN SPLATTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Density functional theory (DFT) strikes a practical balance between accuracy and computational cost in many problems of computational chemistry and materials science. However, many DFT calculations are limited by fixed atom-centered basis sets, which dictate how accuracy and cost scale with system size. We propose *Gaussian Splatting for Density Functional Theory* (GS-DFT), which represents molecular orbitals as a cloud of Gaussians whose positions, shapes, and mixing coefficients are optimized jointly by gradient descent to minimize the energy without training data. Conceptually, GS-DFT is 3D Gaussian splatting with the renderer replaced by quantum mechanics. We introduce two key solver components: adaptive density fitting with screening for efficient evaluation of two-electron integrals, and a regularized differentiable orthogonalization of the molecular orbitals. Empirically, the optimized basis reaches the accuracy of the largest conventional basis sets with a fraction of the parameters, converging systematically in energy, density, and nuclear forces. At equal parameter count, it captures the stretched-bond and anion physics that fixed bases only recover with specialized basis augmentation. The resulting solver exhibits quadratic peak memory scaling in the cloud size, allowing us to simulate systems of up to $2,742$ atoms ($10,406$ electrons) without any modifications at triple-zeta scale using a single four-GPU node.

1 INTRODUCTION

Density functional theory (DFT) offers reliable and practical approximations of many-electron quantum properties from first principles. With a uniquely favorable balance between accuracy and cost, it has matured as a computational pipeline, delivering new scientific insight (Burke, 2012a; Jones, 2015) at scale. Many downstream areas of computational science see DFT as the only interface with quantum phenomena. It is the foundation of modern high-throughput material screening (Jain et al., 2016; Curtarolo et al., 2013; Fiedler et al., 2022), and data generation for machine-learned force fields (Unke et al., 2021b; Jacobs et al., 2025; Hollingsworth & Dror, 2018) in molecular dynamics. At the industrial scale, DFT is used in inverse design settings that require frequent evaluations of quantum mechanical properties (Butler et al., 2018). Routine applications include the conformational energy ranking of drug candidates (Niazi, 2025; Kairys et al., 2019; Guan et al., 2025), characterizing defects and interfaces in semiconductor chip design (Tran & Blaha, 2009; Kim et al., 2006; Van de Walle & Neugebauer, 2004), and mapping reaction pathways or identifying active sites in computational catalysis (Nandy et al., 2021; Sameera & Maseras, 2012; Liao et al., 2022). Therefore, all of these applications inherit DFT's biases and are limited by its data availability.

In practice, calculations in quantum chemistry are bottlenecked by three factors. The first is scalability. Electronic orbitals are expanded in a fixed set of B basis functions, traditionally chosen as approximations of atomic orbitals (AO). The memory cost of the traditional self-consistent field algorithm retains substantial intermediate-storage and tensor contraction costs (Lin & Lu, 2019; Lehtola et al., 2020), making numerical mitigation an area of active research (Becke, 2014). The second is physical fidelity. Accuracy is bounded by how well those predetermined B atomic orbitals can represent the true molecular orbitals. Standard atom-centered basis sets have well-catalogued limitations (Lehtola, 2019; Jensen, 2013; Klopper & Kutzelnigg, 1986) in representing known physics and chemistry without manual fine-tuning. The third is the accuracy of the resulting observables. Orbital basis errors are reflected and often amplified by output quantities, such as forces and densities. Forces converge more slowly with basis size than energies and require additional correction terms (Pulay, 1969) for

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

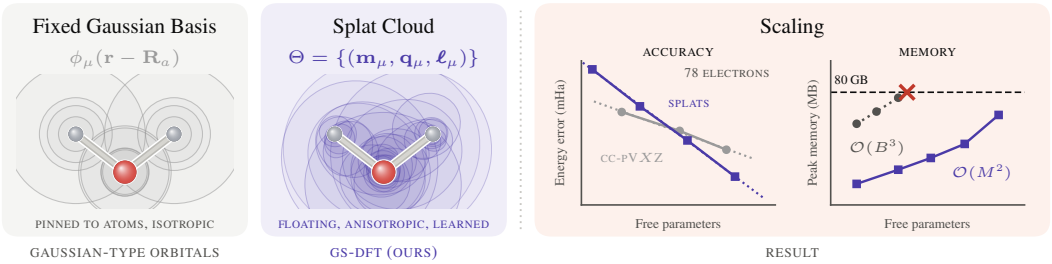

Figure 1: Gaussian Splatting for Density Functional Theory (GS-DFT). Molecular orbitals are expanded in a cloud of M Gaussians (*splats*) whose means and covariances Θ are optimized together with the coefficients $\boldsymbol{C}$ by variational direct energy minimization. The rightmost panels report both scalings for alanine dipeptide (N=78 electrons). The error against the extrapolated references falls faster per free parameter than for the traditional bases with B functions, and peak memory grows as $\mathcal{O}(M^2)$ instead of the $\mathcal{O}(B^3)$ of most traditional codes.

atom-centered bases. The fixed-basis parameterization of electronic orbitals contributes to all three issues. These limitations motivate a solver that is physically constrained by construction, amortizable across related solves, and naturally supports GPU parallelization.

We propose parametrizing electronic orbitals with *floating* anisotropic Gaussian functions. Instead of solving the conventional DFT self-consistency cycle, we directly minimize the total DFT energy using automatic differentiation, yielding a parallelizable direct optimization procedure. Our approach is similar to 3D Gaussian splatting (Kerbl et al., 2023) with the data-driven photometric loss replaced by variational energy optimization. In addition, we propose a memory- and compute-efficient *density fitting* scheme for evaluating the two-electron energy contributions, reducing the peak memory cost from $\mathcal{O}(M^4)$ to $\mathcal{O}(M^2)$ and the per-step tensor contraction cost to $\mathcal{O}(M^2)$ for M Gaussians. The method is summarized in Figure 1. On molecules of increasing size (Section 5), we demonstrate:

- **State-of-the-art accuracy scaling.** We analyze the accuracy scaling laws for GTOs and GS-DFT, showing $2.3\times$ better scaling for all systems. We further show that our results hold for any common exchange-correlation functional type and reproduce the Hartree-Fock ionization potentials.
- **Correct physics without expert corrections.** In regimes where fixed bases have to be augmented by experts (e.g. anions and stretched bonds), GS-DFT reproduces or surpasses the best results of augmented baselines with the augmented budget.
- **Adaptive density fitting for frontier systems.** The *adaptive* density fitting reduces the memory cost to $\mathcal{O}(M^2)$, so the largest system in this work, *protease Nsp1-alpha from the porcine reproductive and respiratory syndrome virus* wiht $\sim 10,000$ electrons, can be optimized on a single node, thus reduceing the memory footprint by a factor of ~ 20.

2 Background

Quantum mechanics as a variational problem. Consider a molecule with N electrons and M nuclei with charges Z_a at positions $\boldsymbol{R}_a \in \mathbb{R}^3$, indexed by $a = 1, \ldots, M$. Its exact quantum properties (the *electronic structure*) are governed by the Hamiltonian

$$\hat{H} = \sum_{i=1}^{N} \left[-\tfrac{1}{2}\nabla^2_{\boldsymbol{r}_i} + v_{\text{ext}}(\boldsymbol{r}_i) \right] + \sum_{i<j} \frac{1}{|\boldsymbol{r}_i - \boldsymbol{r}_j|}, \tag{1}$$

an operator on antisymmetric N-electron wavefunctions $\Psi \in L^2(\mathbb{R}^{3N})$, reflecting physical one- and two-electron interactions. For molecules at equilibrium, $v_{\text{ext}}(\boldsymbol{r}) = -\sum_a Z_a/|\boldsymbol{r} - \boldsymbol{R}_a|$ is the Coulomb attraction to the nuclei. Hamiltonian eigenvalues represent the energies of electronic states. Absent external perturbations, electrons are found in the ground state Ψ_0, the lowest-energy antisymmetric eigenfunction (Sakurai & Napolitano, 2017) of the Hamiltonian in Eq. 1.

Approximating many-electron states by direct minimization of $E[\Psi]$ is a frontier computational problem with prohibitive scaling caused by the exponential growth of the electronic state space

(Medvidović & Moreno, 2024). Kohn-Sham DFT (KS-DFT) (Kohn & Sham, 1965) bypasses this obstacle by modeling a surrogate system of non-interacting electrons that yields the same charge density

$$\rho(\boldsymbol{r}) = N \int \mathrm{d}^3\boldsymbol{r}_2 \cdots \int \mathrm{d}^3\boldsymbol{r}_N \, |\Psi(\boldsymbol{r}, \boldsymbol{r}_2, \ldots, \boldsymbol{r}_N)|^2 \quad \in \; L^1(\mathbb{R}^3) \tag{2}$$

as the full theory given by Eq. 1. KS-DFT is stated as an optimization problem over N single-electron wave functions, or *orbitals*, $\psi_k \in L^2(\mathbb{R}^3)$. Following Eq. 2, the density $\rho(\boldsymbol{r}) = \sum_k |\psi_k(\boldsymbol{r})|^2$ simplifies to a mixture of orbital Born distributions. The KS orbitals are then computed as minimizers of the energy functional $E[\psi]$. The total *approximate* KS energy functional reads $E[\psi] = T_s[\psi] + E_{\text{ext}}[\rho] + E_H[\rho] + E_{\text{xc}}[\psi] + E_{\text{BG}}$ where

$$\begin{aligned} T_s[\psi] &= \frac{1}{2}\sum_{k=1}^{N} \int \mathrm{d}^3\boldsymbol{r} \; |\boldsymbol{\nabla}\psi_k(\boldsymbol{r})|^2 \;, \quad E_{\text{ext}}[\rho] = \int \mathrm{d}^3\boldsymbol{r} \, \rho(\boldsymbol{r}) \, v_{\text{ext}}(\boldsymbol{r}) \;, \\ E_H[\rho] &= \frac{1}{2}\int \mathrm{d}^3\boldsymbol{r} \int \mathrm{d}^3\boldsymbol{r}' \, \frac{\rho(\boldsymbol{r})\rho(\boldsymbol{r}')}{|\boldsymbol{r} - \boldsymbol{r}'|} \;, \quad \text{and} \quad E_{\text{BG}} = \sum_{a<b} \frac{Z_a Z_b}{|\boldsymbol{R}_a - \boldsymbol{R}_b|} \end{aligned} \tag{3}$$

are the kinetic, external, Coulomb (*Hartree*) and background contributions, respectively. The molecular geometry $\{\boldsymbol{R}_a, Z_a\}$ is held fixed under the standard Born-Oppenheimer approximation. The final energy term is the exchange-correlation (XC) energy E_{xc}. The XC energy models all many-electron quantum effects, despite commonly accounting for less than 1% of the total. Therefore, the XC functional has been the central amortized learning problem in DFT for decades (Burke, 2012b; Becke, 2014). With the total energy objective specified, DFT can be formulated as a constrained functional optimization problem

$$\min_{\psi} \; E[\psi] \qquad \text{subject to} \qquad \int \mathrm{d}^3\boldsymbol{r} \; \psi_k(\boldsymbol{r})\psi_l(\boldsymbol{r}) = \delta_{kl} \;. \tag{4}$$

Gaussian-type orbitals. For molecular systems, orbitals are expanded in a fixed basis set of B *Gaussian-type atomic orbital* (GTO) functions χ_μ as $\psi_k(\boldsymbol{r}) = \sum_{\mu=1}^{B} \boldsymbol{C}_{\mu k} \, \chi_\mu(\boldsymbol{r})$. Only the expansion coefficients $\boldsymbol{C}$ are optimized while the Gaussian remains pinned to a given atom, with widths and other shape parameters determined once per element and stored in static tables (Boys, 1950; Ditchfield et al., 1971; Dunning Jr., 1989). Strictly linear combinations of atomic orbitals (LCAO) have remained the default choice because of analytical integration identities and efficient optimization. Firstly, every integral in Eq. 3 has a closed form (Helgaker et al., 2000). Secondly, the linear structure is exploited by the self-consistent field (SCF) fixed-point iteration, which is used as an energy optimization proxy. SCF iterations repeatedly diagonalize a $B \times B$ effective Hamiltonian matrix (Martin, 2004) at $\mathcal{O}(B^3)$ cost and combine the resulting density with the current best ground state estimate (Lin & Lu, 2019).

Two-electron integrals. The LCAO formalism admits pre-computation of intermediate quantities, known as *density fitting* (Dunlap et al., 1979; Dunlap, 2000), enabling fast on-demand two-electron integral estimation for a given system. In DFT, the Hartree energy integral E_H defined in Equation (3) accounts for the classical electrostatic energy of the electronic density ρ. In terms of the fixed basis χ, the density is evaluated through its connection to KS orbitals,

$$\rho(\boldsymbol{r}) = 2\sum_{k} |\psi_k(\boldsymbol{r})|^2 = \sum_{\mu\nu} \boldsymbol{P}_{\mu\nu} \, \chi_\mu(\boldsymbol{r})\chi_\nu(\boldsymbol{r}) \;, \tag{5}$$

with $\boldsymbol{P} = 2\bar{\boldsymbol{C}}\bar{\boldsymbol{C}}^\top$ labeled the *density matrix*. Thus, the Hartree integral can be written out as

$$E_H = \frac{1}{2}\sum_{\mu\nu\lambda\sigma} \boldsymbol{P}_{\mu\nu}\boldsymbol{P}_{\lambda\sigma} \int \mathrm{d}^3\boldsymbol{r} \int \mathrm{d}^3\boldsymbol{r}' \, \frac{\chi_\mu(\boldsymbol{r})\chi_\nu(\boldsymbol{r})\chi_\lambda(\boldsymbol{r}')\chi_\sigma(\boldsymbol{r}')}{|\boldsymbol{r} - \boldsymbol{r}'|} \;. \tag{6}$$

Despite all integrals over GTO products being analytically tractable, naive evaluation of the full electron-repulsion integral (ERI) tensor with $\mathcal{O}(B^4)$ entries in Equation (6) leads to an optimization that costs $\mathcal{O}(B^4)$ in FLOPs and memory.

Density fitting introduces an auxiliary basis as a set of functions $\{\varphi_a\}_{a=1}^{N_{\text{aux}}}$ with $N_{\text{aux}} \ll B^2$. For the Hartree integral in Equation (6), the purpose of the auxiliary basis is to replace the quadratic expansion of the density in terms of the basis χ in Equation (5) with a linear expansion $\rho(\boldsymbol{r}) \approx \sum_{a=1}^{N_{\text{aux}}} \boldsymbol{d}_a \, \varphi_a(\boldsymbol{r})$

in terms of the auxiliary basis. This replacement has the effect of bringing down the degree of the integrand in Equation (6) from quartic to quadratic in χ. The method is called *density fitting* (Dunlap et al., 1979; Dunlap, 2000) because the coefficients $\boldsymbol{d}$ in the auxiliary expansion are determined by a simple least-squares linear fit. The error of this approximation is the Coulomb-metric projection residual of the full quadratic ρ in Equation (5) onto the space spanned by the auxiliary basis, second order in the fit error (Propositions A.4 and A.5). Auxiliary bases for GTOs are human expert-designed GTOs themselves. However, the choice of the auxiliary basis in traditional density fitting introduces a new source of error that has to be independently ablated for each DFT run.

DFT and 3DGS share an optimization problem. In computer vision, 3D Gaussian Splatting (3DGS) (Kerbl et al., 2023) is a state-of-the-art method for 3D scene reconstruction that represents a scene as a linear combination of M anisotropic Gaussian primitives g_{θ_μ}, carrying linear appearance weights c_μ (opacity and color). It minimizes a photometric loss $\mathcal{L}(\cdot,\cdot)$ through a differentiable rendering (*splatting*) operator $\mathcal{R}_v$ against reference views I_v provided as data. The problem

$$\min_{\Theta,\,\boldsymbol{c}} \; \mathop{\mathbb{E}}_{v} \; \mathcal{L}\left(\mathcal{R}_v\left[\textstyle\sum_{\mu=1}^{M} c_\mu\, g_{\theta_\mu}\right],\; I_v\right) \tag{7}$$

is solved jointly in the nonlinear primitive parameters $\Theta = \{\theta_\mu\}_{\mu=1}^{M}$ and the linear weights $\boldsymbol{c}$ for each scene. Writing DFT over the same Gaussian primitives results in a problem similar to Eq. 7, replacing the photometric loss with the total energy functional $E[\cdot]$ from Equation (3). When using GTOs, the conceptual difference between Equations 7 and 4 is only in the constraint. Therefore, DFT is positioned to inherit the entire optimization toolkit that makes Eq. 7 work at scale. While *splatting* in 3DGS refers to density projections onto two-dimensional view planes, DFT optimization can be understood as sequential projections onto the N-dimensional subspace of occupied electronic states.

3 Related Work

Floating and adaptive bases. Floating basis functions began with the floating spherical Gaussian orbitals of Frost (1967), using one Gaussian per electron pair and made anisotropic by ellipsoidal by Vescelius & Frost (1974); Cohen & Basch (1976). Later on, Pederson et al. (1988) minimized the DFT energy over floating Gaussians with Pulay-free forces. However, these approaches remained inaccurate and unscalable due to the lack of a suitable density fitting scheme. Alternatively, materials simulations pursue adaptivity through refinable discretizations instead, finite elements (Schauer & Linder, 2013), wavelets and multiwavelets (Genovese et al., 2008; Jensen et al., 2017), and gausslets (White, 2017), which refine space where needed but multiply degrees of freedom rather than moving them. Most recently, ELECTRA (Elsborg et al., 2025) trains an equivariant network on reference data to predict a molecule's density as a mixture of floating Gaussians that warm-starts a conventional solver, and Medvidović et al. (2026) optimize a superposition of multipole Gaussian splats to compute the optimized effective potential and solve the inverse Kohn-Sham problem.

Machine learning for DFT. Learned components cover the whole pipeline: interatomic potentials replace the calculation (Behler & Parrinello, 2007; Schütt et al., 2017; Batzner et al., 2022; Batatia et al., 2022), learned exchange-correlation functionals improve the energy calculation (Snyder et al., 2012; Dick & Fernandez-Serra, 2020; Chen et al., 2021; Kirkpatrick et al., 2021), and neural networks predict densities, Hamiltonians, and properties in a fixed atomic-orbital basis (Brockherde et al., 2017; Qiao et al., 2020; Unke et al., 2021a; Hazra et al., 2024). Closest to us methodologically is differentiable, direct-optimization DFT: automatic differentiation through Hartree-Fock exposing basis exponents (Tamayo-Mendoza et al., 2018), DQC (Kasim et al., 2022) and D4FT (Li et al., 2023), direct optimization with self-diagonalization in solids (Li et al., 2024), and amortized networks that propose orbitals (Hassan et al., 2025); all of them optimize within, or lightly relax, an atom-centered basis. Instead, we target the basis itself, so every method above composes with GS-DFT.

4 Method

We introduce Gaussian Splatting for Density Functional Theory (GS-DFT), a DFT solver that represents the occupied Kohn-Sham orbitals with a learnable cloud of Gaussian primitives. To efficiently and stably compute the two-electron terms on the fly, we introduce *adaptive* density fitting and a regularized eigensolve to maintain the fundamental orthogonality constraint. The total energy

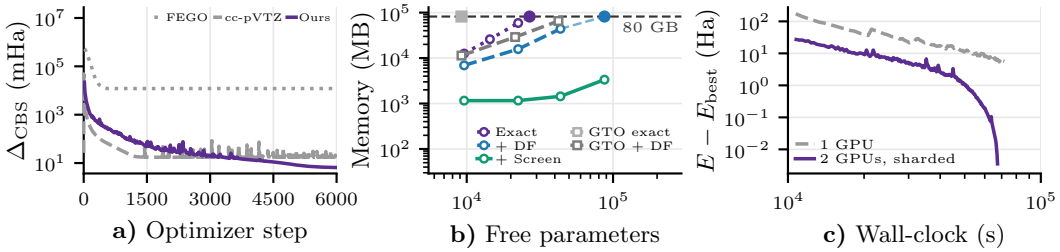

Figure 2: Components of Gaussian Splatting for Density Functional Theory (GS-DFT). *(a) Parametrization of Equation* (8), and the [Floating Elliptical Gaussian Orbitals, FSGO] (Vescelius & Frost, 1974) optimization trajectories against converged GS-DFT of the same function count. *(b) Adaptive density fitting of Equation* (14) with screened, chunked evaluation, plotted as peak memory against free parameters. *(c) The sharded scaling*, with respect to the best energy, against wall-clock time, on chain of 45 alanine residues ($M = 4299$).

functional is optimized using direct variational energy minimization. Figure 2 shows how each ingredient influences GS-DFT scaling compared to traditional solvers.

Floating orbitals as 3D Gaussian Splats. We replace the conventional fixed atom-centered basis with a learnable set of M normalized anisotropic Gaussian primitives, or *splats*,

$$g_\theta(\boldsymbol{r}) = \det\!\left(\tfrac{\boldsymbol{A}}{\pi}\right)^{1/4} \exp\left\{-\tfrac{1}{2}(\boldsymbol{r}-\boldsymbol{m})^\top \boldsymbol{A}\,(\boldsymbol{r}-\boldsymbol{m})\right\}, \tag{8}$$

with center $\boldsymbol{m} \in \mathbb{R}^3$ and symmetric positive-definite precision matrix $\boldsymbol{A} \in \mathbb{R}^{3\times 3}$. Following 3DGS (Kerbl et al., 2023), we parametrize $\boldsymbol{A}$ by its eigendecomposition $\boldsymbol{A} = \boldsymbol{U}(\boldsymbol{q})\mathrm{diag}(e^{\boldsymbol{l}})\boldsymbol{U}(\boldsymbol{q})^\top$, with log-eigenvalues $\boldsymbol{l} \in \mathbb{R}^3$ and a unit quaternion $\boldsymbol{q}$ whose rotation $\boldsymbol{U}(\boldsymbol{q}) \in SO(3)$ orients the principal axes. Every $\theta = \{\boldsymbol{m}, \boldsymbol{l}, \boldsymbol{q}\}$ therefore defines a valid splat with nine free parameters. We index splats with $\mu = 1, \ldots, M$ in arbitrary order, denoting the set of all geometric parameters $\Theta = \{\theta_1, \ldots, \theta_M\}$ and using $g_\mu = g_{\theta_\mu}$ as a shorthand. The N_{occ} occupied orbitals are obtained by linearly mixing these shared primitives, $\psi_k(\boldsymbol{r}) = \sum_\mu \mathbf{C}_{\mu k}\, g_\mu(\boldsymbol{r})$, with the coefficient matrix $\mathbf{C} \in \mathbb{R}^{M\times N_{\mathrm{occ}}}$. Because every splat is positive, the nodal structure of the orbitals comes from nearby splats with coefficients of opposite sign (Propositions B.2 and B.5). Proposition A.1 shows that this parametrization is equivariant under $O(3)$.

Orthogonality constraint. Because the Gaussian primitives in Equation (8) are not generally orthogonal as required by Equation (4), we directly evaluate their overlap matrix $S(\Theta) \in \mathbb{R}^{M\times M}$ as

$$\mathbf{S}_{\mu\nu}(\Theta) = \int \mathrm{d}^3\boldsymbol{r}\; g_\mu(\boldsymbol{r})\, g_\nu(\boldsymbol{r}) = \sqrt{\frac{\det\sqrt{\boldsymbol{A}_\mu \boldsymbol{A}_\nu}}{\det\left(\frac{1}{2}(\boldsymbol{A}_\mu + \boldsymbol{A}_\nu)\right)}}\; \boldsymbol{K}_{\mu\nu} \tag{9}$$

with the matrix $\boldsymbol{K}$ defined as

$$\boldsymbol{K}_{\mu\nu} = \exp\left\{-\tfrac{1}{2}(\boldsymbol{m}_\mu - \boldsymbol{m}_\nu)^\top \boldsymbol{A}_\mu(\boldsymbol{A}_\mu + \boldsymbol{A}_\nu)^{-1}\boldsymbol{A}_\nu(\boldsymbol{m}_\mu - \boldsymbol{m}_\nu)\right\}, \tag{10}$$

so that the orthonormality constraint from Equation (4) reduces to $\mathbf{C}^\top\mathbf{S}(\Theta)\mathbf{C} = I$. We explicitly enforce that constraint in the forward pass via Löwdin orthonormalization,

$$\bar{\boldsymbol{C}} = \boldsymbol{C}\,\boldsymbol{G}^{-1/2} \qquad \text{with} \qquad \boldsymbol{G} = \boldsymbol{C}^\top \boldsymbol{S}(\Theta)\,\boldsymbol{C}, \tag{11}$$

which satisfies $\bar{\boldsymbol{C}}^\top \boldsymbol{S}\bar{\boldsymbol{C}} = I$ identically and is differentiable in both arguments. One backward pass propagates the orthonormality constraint into the basis gradients. We are left with an unconstrained minimization over $\{\boldsymbol{C}, \boldsymbol{m}, \boldsymbol{l}, \boldsymbol{q}\}$, resembling Equation (7). We compute the Löwdin retraction through a *regularized eigendecomposition* $\boldsymbol{G} = \boldsymbol{U}\,\mathrm{diag}(\lambda)\,\boldsymbol{U}^\top$ as

$$\boldsymbol{G}^{-1/2} = \boldsymbol{U}\,\mathrm{diag}\left(f(\lambda)\right)\,\boldsymbol{U}^\top, \qquad \text{where} \qquad f(\lambda) = \max\left(\lambda,\; \varepsilon\,\lambda_{\max}\right)^{-1/2}. \tag{12}$$

Bare gradients of an eigenvalue decomposition are proportional to inverse pairwise eigenvalue differences $(\lambda_i - \lambda_j)^{-1}$, causing numerical instabilities for degenerate matrices. Because of that, we implement a regularized backward pass in which the eigenvalue differences $|\lambda_i - \lambda_j| \le \delta\lambda_{\max}$ are replaced by $\frac{1}{2}(f'(\lambda_i) + f'(\lambda_j))$. Pairs below the floor receive no gradient. Away from degeneracies, the regularization has no effect at convergence.

Adaptive density fitting. Efficient evaluation of the variational energy objective requires fast evaluation of two-electron integrals. The Hartree energy of Equation (6) couples every pair of splat products, so evaluating it exactly costs $O(M^4)$. For Gaussian splats, we extend the static density fitting protocol described in Section 2 with an adaptive auxiliary basis built from the splats themselves. Because the density in Equation (5) is a weighted sum of the splat pair-products, we exploit the fact that each product is a new Gaussian by sub-selecting $N_{\text{aux}} \ll M^2$ of these products to be our auxiliary basis. Specifically, a product of Gaussian splats g_μ and g_ν is an *auxiliary* Gaussian $h_{\mu\nu}$ defined by

$$g_\mu(\boldsymbol{r})\, g_\nu(\boldsymbol{r}) = \boldsymbol{K}_{\mu\nu}\, h_{\mu\nu}(\boldsymbol{r}), \tag{13}$$

where $h_{\mu\nu}$ has precision $\boldsymbol{A}_{\mu\nu} = \boldsymbol{A}_\mu + \boldsymbol{A}_\nu$ and center $\boldsymbol{m}_{\mu\nu} = (\boldsymbol{A}_\mu + \boldsymbol{A}_\nu)^{-1}(\boldsymbol{A}_\mu \boldsymbol{m}_\mu + \boldsymbol{A}_\nu \boldsymbol{m}_\nu)$. The prefactor $\boldsymbol{K}_{\mu\nu}$ is identical to the one in Equation (10) and directly determines which products matter in the expansion in Equation (5). As a result, keeping only the pairs with $|\boldsymbol{S}_{\mu\nu}| > \tau$ is a consistent way of auxiliary basis *screening*, controlled by the hyperparameter $\tau > 0$. For $\tau = 0$, we retain all splat pairs and return to evaluating all $\mathcal{O}(M^4)$ terms in Equation (6). This screening method is related to Schwarz screening (Häser & Ahlrichs, 1989), which is commonly used in large DFT runs to exploit locality. For GS-DFT, it falls out naturally. We find that each splat has $k = \mathcal{O}(1)$ overlap neighbors, resulting in $N_{\text{aux}} = \mathcal{O}(M)$ auxiliary Gaussians in practical calculations.

After specifying the auxiliary basis h_a, it is frozen for K optimization steps and detached from the computation graph. At each of these K steps, the density is expanded as $\rho(\boldsymbol{r}) = \sum_{a=1}^{N_{\text{aux}}} \boldsymbol{d}_a h_a(\boldsymbol{r})$ using the least-squares linear fit onto the density form in Equation (5), where the auxiliary index $a = (\mu, \nu)$ runs over splat pairs. The fit is analytic, resulting in $\boldsymbol{d} = \boldsymbol{Q}^{-1}\boldsymbol{t}$ with closed-form integrals

$$\boldsymbol{Q}_{ab} = \int \mathrm{d}^3\boldsymbol{r} \int \mathrm{d}^3\boldsymbol{r}' \, \frac{h_a(\boldsymbol{r}) h_b(\boldsymbol{r}')}{|\boldsymbol{r} - \boldsymbol{r}'|} \quad \text{and} \quad \boldsymbol{t}_a = \sum_{\mu\nu} \boldsymbol{P}_{\mu\nu} \int \mathrm{d}^3\boldsymbol{r} \int \mathrm{d}^3\boldsymbol{r}' \, \frac{g_\mu(\boldsymbol{r}) g_\nu(\boldsymbol{r}) h_a(\boldsymbol{r}')}{|\boldsymbol{r} - \boldsymbol{r}'|} \tag{14}$$

available because h_a are simple Gaussians themselves. After K gradient updates to primitive Gaussians g_μ, the auxiliary basis is refreshed and the procedure repeats.

Natively differentiable simulation. Every step in the forward map from $(\boldsymbol{C}, \Theta)$ to the energy E is differentiable. This allows us to replace the SCF algorithm with end-to-end optimization by gradient descent or any of its modifications, e.g., Adam (Kingma & Ba, 2015). The splat parametrization being equivariant under O(3) (Proposition A.1), the regularized eigensolve keeping gradients finite at degenerate eigenvalues (Proposition A.6), and the adaptive density fitting bounding the fitting error as the cloud moves (Proposition A.5) all keep this optimization well-behaved. Algorithm 1 gives the full training loop. The auxiliary metric of Equation (14) is factorized once per refresh and reused through K triangular solves, making the dominant cubic operation in the method scale as $\mathcal{O}(N_{\text{aux}}^3/K)$ amortized. We exploit the native differentiability of our simulation in computing forces after the given GS-DFT run has converged. Because no basis function is anchored to an atom, the converged atomic force can be evaluated by direct automatic differentiation. Therefore, the Pulay correction of atom-centered bases (Pulay, 1969) is absent, resulting in a cheaper operation.

Algorithm 1 Gaussian Splatting for Density Functional Theory (GS-DFT).

Require: Nuclei $\{\boldsymbol{R}_a, Z_a\}$; cloud size M; steps T; refresh period K; threshold τ; functional E_{xc}

$\Theta \leftarrow \mathrm{ChemInit}(\boldsymbol{R}, Z)$; $\boldsymbol{C} \leftarrow \mathrm{minAO}(\Theta)$ ▷ Section C.5
for $r = 1, \ldots, \lceil T/K \rceil$ **do**
  $\mathcal{P} \leftarrow \mathrm{stopgrad}\big[\{(\mu,\nu) : |S_{\mu\nu}| > \tau\}\big]$; $\{h_i\}_{i=1}^{N_{\text{aux}}} \leftarrow \mathrm{stopgrad}\big[\mathcal{A}(\Theta)\big]$ ▷ $\mathcal{O}(MN_{\text{aux}})$
  $\boldsymbol{L} \leftarrow \mathrm{stopgrad}\big[\mathrm{chol}(\boldsymbol{Q} + \lambda \mathbb{1})\big]$ ▷ $\mathcal{O}(N_{\text{aux}}^3/K)$
  for $t = 1, \ldots, K$ **do**
    $\boldsymbol{P} \leftarrow 2\,\boldsymbol{C}\,\mathrm{regeig}\big(\boldsymbol{C}^\top \boldsymbol{S}(\Theta)\,\boldsymbol{C}\big)\boldsymbol{C}^\top$ ▷ $\mathcal{O}(M^2 N_{\text{occ}} + N_{\text{occ}}^3)$
    $\boldsymbol{d} \leftarrow \boldsymbol{L}^{-\top}\boldsymbol{L}^{-1}\,\boldsymbol{t}(\Theta, \boldsymbol{P})$ ▷ $\mathcal{O}(MkN_{\text{aux}} + N_{\text{aux}}^2)$
    $E \leftarrow T_s[\boldsymbol{P}] + E_{\text{ext}}[\boldsymbol{P}] + \tilde{E}_H[\boldsymbol{d}] + E_{\text{xc}}[\boldsymbol{P}]$ ▷ $\mathcal{O}(N_g M N_{\text{occ}})$
    $(\Theta, \boldsymbol{C}) \leftarrow \mathrm{Adam}\big((\Theta, \boldsymbol{C}),\ \nabla_{\Theta,\boldsymbol{C}}\, E\big)$
  end for
end for
return $(\Theta, \boldsymbol{C})$; $\boldsymbol{F}_a = -\,\partial E/\partial \boldsymbol{R}_a$ ▷ Proposition A.3

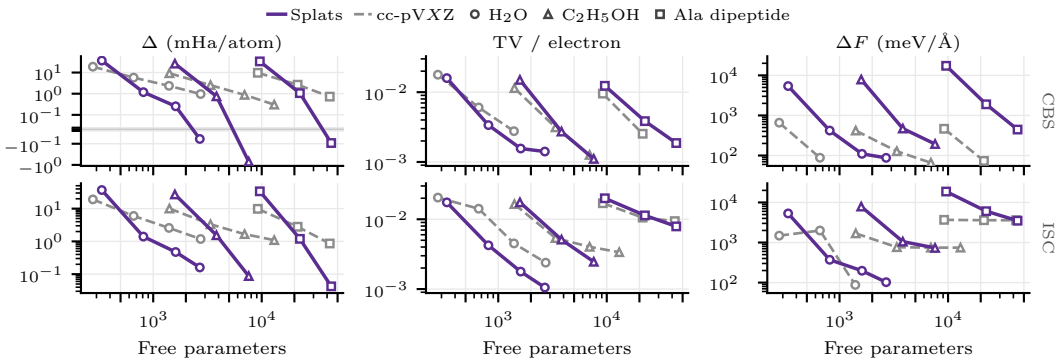

Figure 3: Observables against free parameters. Columns are the energy residual per atom, the total variation of the density per electron, and the maximum deviation of the nuclear forces. **Top row:** measured against the CBS limit. **Bottom row:** measured against the ISC limit.

5 Experiments

In this section, we show three advantages of GS-DFT: accuracy and memory scaling for many tested XC functionals and molecules, diffuse physics representation in stretched molecular geometries, and access to frontier systems using common hardware.

5.1 Accuracy and functional coverage

Accuracy. After optimizing the DFT energy loss, the orbitals are used to make predictions about the quantum-mechanical properties of the molecule. At $\nabla_\Theta E \approx 0$, the energy errors are second-order in small parameter shifts. However, all other downstream observables are first-order sensitive to these small parameter variations (Proposition A.3), requiring numerically precise optimization. There are two main quantities in this category: the electronic density ρ itself and the atomic forces, given by the Hellmann-Feynman derivative $\boldsymbol{F}_a = -\partial E/\partial \boldsymbol{R}_a$ at convergence. To show convergence towards a physical density, we optimize a splat cloud on three benchmark molecules of increasing complexity: water (H_2O), ethanol (C_2H_5OH), and alanine dipeptide ($C_6H_{12}N_2O_2$). We track the three key observables: energy, density, and nuclear forces (Figure 3).

In Figure 3, we show observables relative to extrapolated references as a function of the number of free parameters in GS-DFT, supplemented with GTO results at a matched parameter count. Because DFT is solved as a data-free optimization, unbiased external solutions are difficult to obtain, especially at scale. In Figure 3, we use two types of references: the GTO complete basis set (CBS) limit and the infinite splat cloud (ISC) limit. For any observable O_P computed with a fixed budget of P free parameters, the CBS and ISC limits are independently defined by $P \to \infty$. In practice, we assume that $O_P - O_\infty \propto P^{-\alpha}$ and compute O_∞ and the scaling exponent α by simple regression. We find that GS-DFT consistently produces lower energies in the high-parameter regime in Figure 3. All reported energies in this work are evaluated with full ERIs, eliminating density fitting errors. Density predictions show reliable convergence and similar or better quality than GTO references, as measured by the dimensionless total variation $\mathrm{TV} = \int \mathrm{d}^3\boldsymbol{r}\, |\rho_P(\boldsymbol{r}) - \rho_\infty(\boldsymbol{r})|$. For force predictions, the CBS-ISC distinction is required because the absolute ordering of errors on Figure 3 may depend on the chosen baseline. However, convergence rate α estimates remain similar. A third, independently discretized and converged accurate method would resolve absolute force accuracy.

Functional coverage. The splat representation of KS orbitals is independent of the choice of the XC functional. We test this on ethanol across four rungs of functional complexity: PBE (Perdew et al., 1996) (a generalized gradient approximation, GGA), r^{2}SCAN (Furness et al., 2020) (a meta-GGA), the global hybrid B3LYP (Becke, 1993), and the range-separated hybrid ωB97M-V (Mardirossian & Head-Gordon, 2016). In Table 1, we compare a GTO reference, cc-pVTZ basis, to GS-DFT at matched function count, $N_{\mathrm{ao}} = M = 174$. Energies achieved by GS-DFT sit closer to the CBS limit than the GTO reference for all four functionals. For the widely-used PBE and r^{2}SCAN functionals, we achieve optimized energy values in the range of the cc-pV5Z GTO basis, using 29% of its functions. In addition, reliable convergence and predictions across a wide variety of XC approximations suggest that GS-DFT predicts physical densities instead of exploiting numerical pathologies in the non-linear optimization with adaptive density fitting.

Table 1: Functional Coverage. Ethanol's Δ_{CBS} with $M = n_{\text{ao}} = 174$. %M denotes the percentage of M used by our method.

	$\Delta_{\text{CBS}} \downarrow$		
Functional	GS-DFT	cc-pVTZ	**%M**
PBE	**2.3**	22.8	< 29
B3LYP	**16.9**	21.3	77
r^{2}SCAN	**1.8**	19.9	< 29
ωB97M-V	**10.2**	19.8	61

Table 2: Ionization potential at matched function count. Under Hartree–Fock, $-\varepsilon_{\text{HOMO}}$ *is* the ionization potential, in eV. GS-DFT uses $\mathbf{M} = n_{\text{ao}}$ of def2-QZVPP; the references are the CCSD(T) and experimental values of GW100.

		$-\varepsilon_{\text{HOMO}}$		**Reference (GW100)**	
System	$\mathbf{M} = n_{\text{ao}}$	GS-DFT	def2-QZVPP	CCSD(T)	Exp.
Li_2	70	4.950	4.950	5.266	4.73
H_2O	117	13.878	13.870	12.565	12.62
C_2H_5OH	351	12.042	12.040	10.685	10.64
C_6H_6	522	9.148	9.149	9.29	9.24

Ionization Potential. The negative of the highest occupied orbital energy equals the ionization potential (IP) in the Hartree-Fock (HF) approximation (Koopmans, 1934). We treat HF as a special case of DFT, with an energy functional in Equation (3) set to include exchange and ignore correlation. We evaluate the largest orbital eigenvalue $\varepsilon_{\text{HOMO}}$ against the GW100 (van Setten et al., 2015), a high-precision benchmark of one hundred closed-shell molecules whose IPs are computed with coupled cluster with singles, doubles, perturbative triples (CCSD(T)). We further add the measured experimental values to compare against (Maggio et al., 2017). At matched function-count, GS-DFT reproduces the GTO references across all systems, as shown in Table 2. The remaining error of both GS-DFT and GTOs with respect to CCSD(T) and experimental values is attributed to orbital relaxation and the absence of electronic correlation effects. In addition to the energy, the density, and the forces, the IP serves as an independent numerical test against high-precision numerics and measurements.

5.2 Splats as a Physical Representation

Narrow conventional basis sets anchored at nuclei can lack the support needed to describe density that extends far from them. In bare anions, the electron density expands away from the nuclei because of the extra electron-electron repulsion. Traditionally, this problem in anions and stretched bonds has been mitigated by *augmenting* basis sets with extra diffuse fixed Gaussians, e.g., the aug-cc-pVζZ families (Kendall et al., 1992). This augmentation needs to be triggered by human experts before a calculation is started, as GTOs cannot automatically expand their support during optimization. Splats have adaptive support and do not require expert intervention to represent diffuse systems. In this subsection, we test GS-DFT on anions with long density tails and stretched chemical bonds.

The left plot in Figure 4 shows how the splats converge to better energy values than the augmented baseline, cc-pVTZ. In GS-DFT computations in Figure 4, we use the same number of functions as the unaugmented basis set. For stretched bond geometries, electronic densities can maintain nonzero values in the interstitial region. In those cases, GTOs cannot extend their support to faithfully represent relevant delocalized orbitals. In the right plot in Figure 4, we show this behavior by stretching lithium fluoride (LiF). We show that splats with adaptive support achieve lower energies compared to the GTO calculations with both cc-pVTZ and the manually augmented aug-cc-pVTZ variant, while using the unaugmented number of basis functions. Additionally, direct energy optimization in GS-DFT has shown robust convergence properties for all considered geometries, while SCF has displayed well-known convergence problems for highly deformed molecules. See Section F.2 for details.

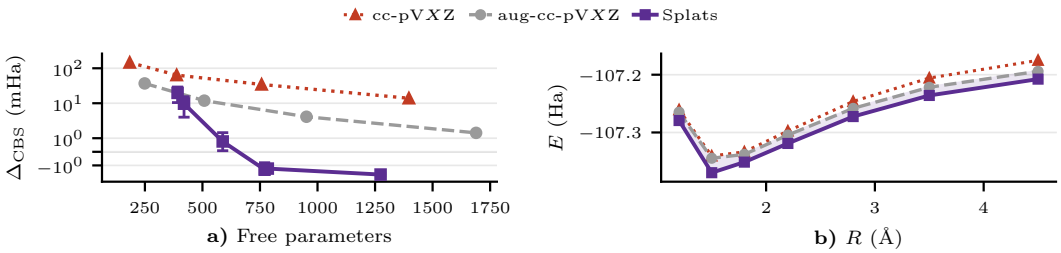

Figure 4: (a) Fluoride ion (F^-) basis-set convergence. $\Delta_{\text{CBS}} = E - E_{\text{CBS}}$ is the residual from the CBS limit in milli-Hartree (mHa). **(b) LiF dissociation at matched count.** The margin over the unaugmented basis grows as the bond stretches, while the margin over the augmented basis stays flat.

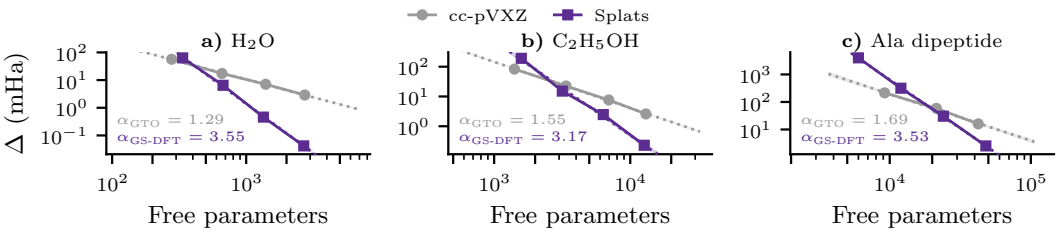

Figure 5: Accuracy scaling law for GS-DFT. GTO rungs against the extrapolated complete basis set (CBS) energy, and the splat clouds against the infinite splat cloud (ISC) energy. Limits are obtained by fitting $\Delta = E_P - E_\infty \propto P^{-\alpha}$ against four runs with P parameters.

5.3 Accuracy and Memory Scaling

Finally, we examine two notions of scaling for GS-DFT: how rapidly basis-set errors decrease with representation expressivity and how accuracy and the memory footprint grow with molecule size.

Accuracy scaling law. We study the scaling of the energy error with the number of free parameters P in Figure 5. We define the energy error and its scaling power law as $\Delta = E - E_\infty \propto P^{-\alpha}$, independently for GTOs and splats. Our analysis, in Figure 5, indicates that GS-DFT can achieve $2.3\times$ sharper scaling exponents on average. Across all systems studied in this work, we observe similar scaling improvements. This observation supports the intuition that using splats as an *adaptive* basis is more parameter efficient because scaling is determined by the number of basis functions, regardless of how each function is parametrized. Therefore, packing more expressivity into each function is an efficient way to gain accuracy with minimal scalability cost.

Memory scaling. Scaling DFT beyond hundreds of atoms requires careful optimization, parallelization, and high-performance computing (HPC) using linear-scaling methods (Goedecker, 1999; Bowler & Miyazaki, 2012; Kitaura et al., 1999) that trade off accuracy for scale (Prodan & Kohn, 2005). To show memory scalability, we run five systems from the Fragment Molecular Orbital (FMO) database (Takaya et al., 2021; Watanabe et al., 2019), and compare our GS-DFT results with their references, calculated with the linear-scaling ABINIT-MP package (Tanaka et al., 2014) implementing Hartree-Fock (HF) with 6-31g(d) basis using state-of-the-art HPC resources (Doi et al., 2025). In contrast, we tackle the same large systems with the PBE functional, using a fraction of the memory estimated for the density-fitted cc-pVTZ GTO implementation and inter-node communication, see Table 3. GS-DFT's favorable memory scaling and data-parallel direct minimization allow us to take advantage of modern hardware and codes used in distributed AI training.

Table 3: FMO systems. Energy in Ha, memory in GB. One $4\times$H200 GPU node has 564 GB in total.

			Peak Mem.				Fits GPU node		Simple forces	
System	**Atoms**	**M**	GS-DFT	GTO	$E_{GS\text{-}DFT}$	E_{HF}	GS-DFT	GTO	GS-DFT	GTO
Trp-cage	304	6,720	25.2	640*	−7,478.3	−7,439.0	✓	✗	✓	✗
Defensin	465	10,390	39.6	1,060*	−13,685.4	−13,624.0	✓	✗	✓	✗
Insulin	947	21,026	92.8	2,400*	−25,523.2	−25,404.2	✓	✗	✓	✗
β-spectrin CH	1,755	38,794	212.0	4,850*	−44,255.4	−44,045.7	✓	✗	✓	✗
3IFU (chain A)	2,742	60,308	479.6	8,070*	−69,823.2	−69,631.4	✓	✗	✓	✗

*Density-fitted cc-pVTZ, extrapolated from measured runs.

6 Conclusion

Every DFT calculation is currently limited by its fixed basis set. We propose GS-DFT, expanding the orbitals in a cloud of floating Gaussians and optimizing their parameters by gradient descent on the variational energy loss. This enables an *adaptive* representation that achieves lower energies, delivers correct observables, and automatically captures diffuse physics. Its memory scaling brings frontier quantum systems within reach of modern hardware accelerators. However, GS-DFT requires more optimization steps and thus more time to converge. We see this constraint as an opportunity for future research, e.g., physics-informed natural gradient optimization. Numerical cost is also amortizable by recent methods at the interface of foundation models and 3DGS, taking advantage of a uniquely parameter- and memory-efficient representation of quantum physics offered by GS-DFT.

Reproducibility Statement

The code is available in an anonymized repository at `https://anonymous.4open.science/r/gs-dft-26FC`. It contains the implementation, the configuration and launch script of every experiment, the data every figure is built from, and every molecular geometry except the FMO proteins, which are available from FMODB (Takaya et al., 2021). The proofs of the theoretical results are in Section A. Section C gives the integrals, the training settings (Table 8), the evaluation protocol, and the hardware and software versions.

AI Use Statement

We used large language models to assist with writing and debugging code, with formatting the manuscript, and with finding relevant literature. The authors designed the method and the experiments, checked the code, the results and every cited work, and take full responsibility for the content of the paper.

Impact Statement

Gaussian Splatting for Density Functional Theory is a solver for *ab-initio* molecule simulation. Its consequences are those of better DFT tooling: efficient quantum-mechanical data generation with a memory footprint that keeps larger systems on common hardware. We foresee no societal risks specific to this method beyond those of computational chemistry at large.

References

Ilyes Batatia, Dávid Péter Kovács, Gregor N. C. Simm, Christoph Ortner, and Gábor Csányi. MACE: Higher order equivariant message passing neural networks for fast and accurate force fields. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Simon Batzner, Albert Musaelian, Lixin Sun, Mario Geiger, Jonathan P. Mailoa, Mordechai Kornbluth, Nicola Molinari, Tess E. Smidt, and Boris Kozinsky. E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nature Communications*, 13:2453, 2022.

Atilim Gunes Baydin, Barak A. Pearlmutter, Alexey Andreyevich Radul, and Jeffrey Mark Siskind. Automatic Differentiation in Machine Learning: A Survey. *Journal of Machine Learning Research*, 18(153):1–43, 2018. ISSN 1533-7928.

Axel D. Becke. Density-functional thermochemistry. III. The role of exact exchange. *The Journal of Chemical Physics*, 98(7):5648–5652, April 1993. ISSN 0021-9606. doi: 10.1063/1.464913.

Axel D. Becke. Perspective: Fifty years of density-functional theory in chemical physics. *The Journal of Chemical Physics*, 140(18):18A301, April 2014. ISSN 0021-9606. doi: 10.1063/1.4869598.

Jörg Behler and Michele Parrinello. Generalized neural-network representation of high-dimensional potential-energy surfaces. *Physical Review Letters*, 98(14):146401, 2007.

Rajendra Bhatia. *Matrix Analysis*, volume 169 of *Graduate Texts in Mathematics*. Springer New York, 1997. doi: 10.1007/978-1-4612-0653-8.

David R. Bowler and Tsuyoshi Miyazaki. $\mathcal{O}(N)$ methods in electronic structure calculations. *Reports on Progress in Physics*, 75(3):036503, 2012. doi: 10.1088/0034-4885/75/3/036503.

S. Francis Boys. Electronic wave functions. I. A general method of calculation for the stationary states of any molecular system. *Proceedings of the Royal Society of London. Series A*, 200(1063): 542–554, 1950.

James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: Composable transformations of Python+NumPy programs. `http://github.com/google/jax`, 2018.

Felix Brockherde, Leslie Vogt, Li Li, Mark E. Tuckerman, Kieron Burke, and Klaus-Robert Müller. Bypassing the Kohn-Sham equations with machine learning. *Nature Communications*, 8:872, 2017.

Kieron Burke. Perspective on density functional theory. *The Journal of Chemical Physics*, 136(15): 150901, 2012a.

Kieron Burke. Perspective on density functional theory. *The Journal of Chemical Physics*, 136(15), April 2012b. ISSN 0021-9606, 1089-7690. doi: 10.1063/1.4704546.

Keith T. Butler, Daniel W. Davies, Hugh Cartwright, Olexandr Isayev, and Aron Walsh. Machine learning for molecular and materials science. *Nature*, 559(7715):547–555, July 2018. ISSN 1476-4687. doi: 10.1038/s41586-018-0337-2.

Yixiao Chen, Linfeng Zhang, Han Wang, and Weinan E. DeePKS: A comprehensive data-driven approach toward chemically accurate density functional theory. *Journal of Chemical Theory and Computation*, 17(1):170–181, 2021.

Maurice Cohen and Harold Basch. The ellipsoidal Gaussian basis in molecular orbital theory. *Theoretica Chimica Acta*, 42(3):199–206, 1976.

Stefano Curtarolo, Gus L. W. Hart, Marco Buongiorno Nardelli, Natalio Mingo, Stefano Sanvito, and Ohad Levy. The high-throughput highway to computational materials design. *Nature Materials*, 12(3):191–201, March 2013. ISSN 1476-1122, 1476-4660. doi: 10.1038/nmat3568.

Sebastian Dick and Marivi Fernandez-Serra. Machine learning accurate exchange and correlation functionals of the electronic density. *Nature Communications*, 11:3509, 2020.

R. Ditchfield, W. J. Hehre, and John A. Pople. Self-consistent molecular-orbital methods. IX. An extended Gaussian-type basis for molecular-orbital studies of organic molecules. *The Journal of Chemical Physics*, 54(2):724–728, 1971.

Hideo Doi, Tatsuya Nakano, Kota Sakakura, Yuji Mochizuki, Koji Okuwaki, Toshihiro Kato, Hiroyuki Takizawa, Satoshi Ohshima, Tetsuya Hoshino, and Takahiro Katagiri. Large-scale FMO-MP2 calculations of the spike protein droplet model. *Journal of Computational Chemistry*, 46(4):e70052, 2025. doi: 10.1002/jcc.70052.

Brett I. Dunlap. Robust and variational fitting. *Physical Chemistry Chemical Physics*, 2(10):2113–2116, 2000.

Brett I. Dunlap, J. W. D. Connolly, and John R. Sabin. On some approximations in applications of $X\alpha$ theory. *The Journal of Chemical Physics*, 71(8):3396–3402, 1979.

Thom H. Dunning Jr. Gaussian basis sets for use in correlated molecular calculations. I. The atoms Boron through Neon and Hydrogen. *The Journal of Chemical Physics*, 90(2):1007–1023, 1989.

Jonas Elsborg, Luca Thiede, Alán Aspuru-Guzik, Tejs Vegge, and Arghya Bhowmik. ELECTRA: A Cartesian network for 3D charge density prediction with floating orbitals. In *Advances in Neural Information Processing Systems*, 2025.

R. P. Feynman. Forces in molecules. *Physical Review*, 56(4):340–343, 1939. doi: 10.1103/PhysRev.56.340.

L. Fiedler, K. Shah, M. Bussmann, and A. Cangi. Deep dive into machine learning density functional theory for materials science and chemistry. *Physical Review Materials*, 6(4):040301, April 2022. ISSN 2475-9953. doi: 10.1103/PhysRevMaterials.6.040301.

Arthur A. Frost. Floating spherical Gaussian orbital model of molecular structure. I. Computational procedure. LiH as an example. *The Journal of Chemical Physics*, 47(10):3707–3713, 1967.

James W. Furness, Aaron D. Kaplan, Jinliang Ning, John P. Perdew, and Jianwei Sun. Accurate and Numerically Efficient r ^{2} SCAN Meta-Generalized Gradient Approximation. *The Journal of Physical Chemistry Letters*, 11(19):8208–8215, October 2020. ISSN 1948-7185, 1948-7185. doi: 10.1021/acs.jpclett.0c02405.

Luigi Genovese, Alexey Neelov, Stefan Goedecker, Thierry Deutsch, Seyed Alireza Ghasemi, Alexander Willand, Damien Caliste, Oded Zilberberg, Mark Rayson, Anders Bergman, and Reinhold Schneider. Daubechies wavelets as a basis set for density functional pseudopotential calculations. *The Journal of Chemical Physics*, 129(1):014109, 2008.

Stefan Goedecker. Linear scaling electronic structure methods. *Reviews of Modern Physics*, 71(4): 1085–1123, 1999. doi: 10.1103/RevModPhys.71.1085.

Haoyue Guan, Huimin Sun, and Xia Zhao. Application of Density Functional Theory to Molecular Engineering of Pharmaceutical Formulations. *International Journal of Molecular Sciences*, 26(7), April 2025. ISSN 1422-0067. doi: 10.3390/ijms26073262.

Andrés Guzmán-Cordero. dftax: a differentiable Kohn-Sham DFT engine in JAX, 2026.

Marco Häser and Reinhart Ahlrichs. Improvements on the direct SCF method. *Journal of Computational Chemistry*, 10(1):104–111, 1989. doi: 10.1002/jcc.540100111.

Majdi Hassan, Cristian Gabellini, Hatem Helal, Dominique Beaini, and Kirill Neklyudov. Self-refining training for amortized density functional theory, 2025.

S. Hazra, U. Patil, and S. Sanvito. Predicting the one-particle density matrix with machine learning. *Journal of Chemical Theory and Computation*, 20, 2024.

Hatem Helal and Andrew Fitzgibbon. MESS: Modern electronic structure simulations. *arXiv preprint arXiv:2406.03121*, 2024.

Trygve Helgaker, Poul Jørgensen, and Jeppe Olsen. *Molecular Electronic-Structure Theory*. Wiley, 2000.

Scott A. Hollingsworth and Ron O. Dror. Molecular Dynamics Simulation for All. *Neuron*, 99(6): 1129–1143, September 2018. ISSN 08966273. doi: 10.1016/j.neuron.2018.08.011.

A. C. Hurley. The electrostatic calculation of molecular energies. I. Methods of calculating molecular energies. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 226(1165):170–178, 1954. doi: 10.1098/rspa.1954.0246.

Ryan Jacobs, Dane Morgan, Siamak Attarian, Jun Meng, Chen Shen, Zhenghao Wu, Clare Yijia Xie, Julia H. Yang, Nongnuch Artrith, Ben Blaiszik, Gerbrand Ceder, Kamal Choudhary, Gabor Csanyi, Ekin Dogus Cubuk, Bowen Deng, Ralf Drautz, Xiang Fu, Jonathan Godwin, Vasant Honavar, Olexandr Isayev, Anders Johansson, Boris Kozinsky, Stefano Martiniani, Shyue Ping Ong, Igor Poltavsky, K. J. Schmidt, So Takamoto, Aidan Thompson, Julia Westermayr, and Brandon M. Wood. A practical guide to machine learning interatomic potentials – Status and future. *Current Opinion in Solid State and Materials Science*, 35:101214, March 2025. ISSN 13590286. doi: 10.1016/j.cossms.2025.101214.

Anubhav Jain, Yongwoo Shin, and Kristin A. Persson. Computational predictions of energy materials using density functional theory. *Nature Reviews Materials*, 1(1):15004, January 2016. ISSN 2058-8437. doi: 10.1038/natrevmats.2015.4.

Frank Jensen. Atomic orbital basis sets. *WIREs Computational Molecular Science*, 3(3):273–295, 2013. ISSN 1759-0884. doi: 10.1002/wcms.1123.

Stig Rune Jensen, Santanu Saha, José A. Flores-Livas, William Huhn, Volker Blum, Stefan Goedecker, and Luca Frediani. The elephant in the room of density functional theory calculations. *The Journal of Physical Chemistry Letters*, 8(7):1449–1457, 2017.

R. O. Jones. Density functional theory: Its origins, rise to prominence, and future. *Reviews of Modern Physics*, 87(3):897–923, 2015.

Visvaldas Kairys, Lina Baranauskiene, Migle Kazlauskiene, Daumantas Matulis, and Egidijus Kazlauskas. Binding affinity in drug design: Experimental and computational techniques. *Expert Opinion on Drug Discovery*, 14(8):755–768, August 2019. ISSN 1746-0441. doi: 10.1080/17460441.2019.1623202.

Muhammad F. Kasim, Susi Lehtola, and Sam M. Vinko. DQC: A Python program package for differentiable quantum chemistry. *The Journal of Chemical Physics*, 156(8):084801, 2022.

Rick A. Kendall, Thom H. Dunning Jr., and Robert J. Harrison. Electron affinities of the first-row atoms revisited. Systematic basis sets and wave functions. *The Journal of Chemical Physics*, 96(9): 6796–6806, 1992.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D Gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4):139:1–139:14, 2023.

Patrick Kidger and Cristian Garcia. Equinox: Neural networks in JAX via callable PyTrees and filtered transformations, 2021.

D. C. Kim, M. J. Lee, S. E. Ahn, S. Seo, J. C. Park, I. K. Yoo, I. G. Baek, H. J. Kim, E. K. Yim, J. E. Lee, S. O. Park, H. S. Kim, U-In Chung, J. T. Moon, and B. I. Ryu. Improvement of resistive memory switching in NiO using IrO2. *Applied Physics Letters*, 88(23):232106, June 2006. ISSN 0003-6951, 1077-3118. doi: 10.1063/1.2210087.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

James Kirkpatrick, Brendan McMorrow, David H. P. Turban, Alexander L. Gaunt, James S. Spencer, Alexander G. D. G. Matthews, Annette Obika, Louis Thiry, Meire Fortunato, David Pfau, et al. Pushing the frontiers of density functionals by solving the fractional electron problem. *Science*, 374(6573):1385–1389, 2021.

Kazuo Kitaura, Eiji Ikeo, Toshio Asada, Tatsuya Nakano, and Masami Uebayasi. Fragment molecular orbital method: an approximate computational method for large molecules. *Chemical Physics Letters*, 313(3-4):701–706, 1999. doi: 10.1016/S0009-2614(99)00874-X.

Wim Klopper and Werner Kutzelnigg. Gaussian basis sets and the nuclear cusp problem. *Journal of Molecular Structure: THEOCHEM*, 135:339–356, 1986.

W. Kohn and L. J. Sham. Self-consistent equations including exchange and correlation effects. *Physical Review*, 140(4A):A1133–A1138, 1965.

T Koopmans. Über die Zuordnung von Wellenfunktionen und Eigenwerten zu den Einzelnen Elektronen Eines Atoms. *Physica*, 1(1-6):104–113, January 1934. ISSN 00318914. doi: 10.1016/S0031-8914(34)90011-2.

Susi Lehtola. A review on non-relativistic, fully numerical electronic structure calculations on atoms and diatomic molecules. *International Journal of Quantum Chemistry*, 119(19):e25968, 2019. ISSN 1097-461X. doi: 10.1002/qua.25968.

Susi Lehtola, Frank Blockhuys, and Christian Van Alsenoy. An Overview of Self-Consistent Field Calculations Within Finite Basis Sets. *Molecules*, 25(5):1218, March 2020. ISSN 1420-3049. doi: 10.3390/molecules25051218.

Tianbo Li, Min Lin, Zheyuan Hu, Kunhao Zheng, Giovanni Vignale, Kenji Kawaguchi, A. H. Castro Neto, Kostya S. Novoselov, and Shuicheng Yan. D4FT: A deep learning approach to Kohn-Sham density functional theory. In *International Conference on Learning Representations*, 2023.

Tianbo Li, Min Lin, Stephen G. Dale, Zekun Shi, A. H. Castro Neto, Kostya S. Novoselov, and Giovanni Vignale. Diagonalization without diagonalization: A direct optimization approach for solid-state density functional theory. *arXiv preprint arXiv:2411.05033*, 2024.

Xiaobin Liao, Ruihu Lu, Lixue Xia, Qian Liu, Huan Wang, Kristin Zhao, Zhaoyang Wang, and Yan Zhao. Density Functional Theory for Electrocatalysis. *ENERGY & ENVIRONMENTAL MATERIALS*, 5(1):157–185, 2022. ISSN 2575-0356. doi: 10.1002/eem2.12204.

Lin Lin and Jianfeng Lu. *A Mathematical Introduction to Electronic Structure Theory*. Society for Industrial and Applied Mathematics, Philadelphia, PA, January 2019. ISBN 978-1-61197-579-6 978-1-61197-580-2. doi: 10.1137/1.9781611975802.

Emanuele Maggio, Peitao Liu, Michiel J. van Setten, and Georg Kresse. GW100: A plane wave perspective for small molecules. *Journal of Chemical Theory and Computation*, 13(2):635–648, 2017. doi: 10.1021/acs.jctc.6b01150.

Narbe Mardirossian and Martin Head-Gordon. ωb97m-V: A combinatorially optimized, range-separated hybrid, meta-GGA density functional with VV10 nonlocal correlation. *The Journal of Chemical Physics*, 144(21):214110, June 2016. ISSN 0021-9606, 1089-7690. doi: 10.1063/1.4952647.

Richard M. Martin. *Electronic Structure: Basic Theory and Practical Methods*. Cambridge University Press, 2004.

Matija Medvidović and Javier Robledo Moreno. Neural-network quantum states for many-body physics. *The European Physical Journal Plus*, 139(7):631, July 2024. ISSN 2190-5444. doi: 10.1140/epjp/s13360-024-05311-y.

Matija Medvidović, Angel Rubio, and Juan Carrasquilla. Multipole splats for optimized and inverted effective potentials, 2026. URL https://arxiv.org/abs/2609.09280.

Jim Mitroy, Sergiy Bubin, Wataru Horiuchi, Yasuyuki Suzuki, Ludwik Adamowicz, Wojciech Cencek, Krzysztof Szalewicz, Jacek Komasa, D. Blume, and Kálmán Varga. Theory and application of explicitly correlated Gaussians. *Reviews of Modern Physics*, 85(2):693–749, 2013.

Aditya Nandy, Chenru Duan, Michael G. Taylor, Fang Liu, Adam H. Steeves, and Heather J. Kulik. Computational Discovery of Transition-metal Complexes: From High-throughput Screening to Machine Learning. *Chemical Reviews*, 121(16):9927–10000, August 2021. ISSN 0009-2665. doi: 10.1021/acs.chemrev.1c00347.

Sarfaraz K. Niazi. Quantum Mechanics in Drug Discovery: A Comprehensive Review of Methods, Applications, and Future Directions. *International Journal of Molecular Sciences*, 26(13):6325, June 2025. ISSN 1422-0067. doi: 10.3390/ijms26136325.

Mark R. Pederson, Barry M. Klein, and Jeremy Q. Broughton. Simulated annealing with floating Gaussians: Hellmann-Feynman forces without corrections. *Physical Review B*, 38(6):3825–3833, 1988.

John P. Perdew, Kieron Burke, and Matthias Ernzerhof. Generalized gradient approximation made simple. *Physical Review Letters*, 77(18):3865–3868, 1996.

Emil Prodan and Walter Kohn. Nearsightedness of electronic matter. *Proceedings of the National Academy of Sciences*, 102(33):11635–11638, 2005. doi: 10.1073/pnas.0505436102.

Péter Pulay. Ab initio calculation of force constants and equilibrium geometries in polyatomic molecules. I. Theory. *Molecular Physics*, 17(2):197–204, 1969.

Zhuoran Qiao, Matthew Welborn, Animashree Anandkumar, Frederick R. Manby, and Thomas F. Miller III. OrbNet: Deep learning for quantum chemistry using symmetry-adapted atomic-orbital features. *The Journal of Chemical Physics*, 153(12):124111, 2020.

Jun J. Sakurai and Jim Napolitano. *Modern Quantum Mechanics*. Cambridge University Press, Cambridge, second edition edition, 2017. ISBN 978-1-108-42241-3.

W. M. C. Sameera and Feliu Maseras. Transition metal catalysis by density functional theory and density functional theory/molecular mechanics. *WIREs Computational Molecular Science*, 2(3):375–385, 2012. ISSN 1759-0884. doi: 10.1002/wcms.1092.

Volker Schauer and Christian Linder. All-electron Kohn-Sham density functional theory on hierarchic finite element spaces. *Journal of Computational Physics*, 250:644–664, 2013.

Kristof T. Schütt, Pieter-Jan Kindermans, Huziel E. Sauceda, Stefan Chmiela, Alexandre Tkatchenko, and Klaus-Robert Müller. SchNet: A continuous-filter convolutional neural network for modeling quantum interactions. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

John C. Snyder, Matthias Rupp, Katja Hansen, Klaus-Robert Müller, and Kieron Burke. Finding density functionals with machine learning. *Physical Review Letters*, 108(25):253002, 2012.

Qiming Sun, Xing Zhang, Samragni Banerjee, Peng Bao, Marc Barbry, Nick S. Blunt, Nikolay A. Bogdanov, George H. Booth, Jia Chen, Zhi-Hao Cui, Janus J. Eriksen, Yang Gao, Sheng Guo, Jan Hermann, Matthew R. Hermes, Kevin Koh, Peter Koval, Susi Lehtola, Zhendong Li, Junzi Liu, Narbe Mardirossian, James D. McClain, Mario Motta, Bastien Mussard, Hung Q. Pham, Artem Pulkin, Wirawan Purwanto, Paul J. Robinson, Enrico Ronca, Elvira R. Sayfutyarova, Maximilian Scheurer, Henry F. Schurkus, James E. T. Smith, Chong Sun, Shi-Ning Sun, Shiv Upadhyay, Lucas K. Wagner, Xiao Wang, Alec White, James Daniel Whitfield, Mark J. Williamson, Sebastian Wouters, Jun Yang, Jason M. Yu, Tianyu Zhu, Timothy C. Berkelbach, Sandeep Sharma, Alexander Yu. Sokolov, and Garnet Kin-Lic Chan. Recent developments in the PySCF program package. *The Journal of Chemical Physics*, 153(2):024109, 2020.

Daisuke Takaya, Chiduru Watanabe, Shunpei Nagase, Kikuko Kamisaka, Yoshio Okiyama, Hirotomo Moriwaki, Hitomi Yuki, Tomohiro Sato, Noriyuki Kurita, Yoichiro Yagi, Tatsuya Takagi, Norihito Kawashita, Kenichiro Takaba, Tomonaga Ozawa, Midori Takimoto-Kamimura, Shigenori Tanaka, Kaori Fukuzawa, and Teruki Honma. FMODB: The World's First Database of Quantum Mechanical Calculations for Biomacromolecules Based on the Fragment Molecular Orbital Method. *Journal of Chemical Information and Modeling*, 61(2):777–794, 2021. doi: 10.1021/acs.jcim.0c01062.

Teresa Tamayo-Mendoza, Christoph Kreisbeck, Roland Lindh, and Alán Aspuru-Guzik. Automatic differentiation in quantum chemistry with applications to fully variational Hartree-Fock. *ACS Central Science*, 4(5):559–566, 2018.

Shigenori Tanaka, Yuji Mochizuki, Yuto Komeiji, Yoshio Okiyama, and Kaori Fukuzawa. Electron-correlated fragment-molecular-orbital calculations for biomolecular and nano systems. *Physical Chemistry Chemical Physics*, 16(22):10310–10344, 2014. doi: 10.1039/C4CP00316K.

Fabien Tran and Peter Blaha. Accurate Band Gaps of Semiconductors and Insulators with a Semilocal Exchange-Correlation Potential. *Physical Review Letters*, 102(22):226401, June 2009. ISSN 0031-9007, 1079-7114. doi: 10.1103/PhysRevLett.102.226401.

Oliver T. Unke, Mihail Bogojeski, Michael Gastegger, Mario Geiger, Tess Smidt, and Klaus-Robert Müller. SE(3)-equivariant prediction of molecular wavefunctions and electronic densities. In *Advances in Neural Information Processing Systems*, volume 34, 2021a.

Oliver T. Unke, Stefan Chmiela, Huziel E. Sauceda, Michael Gastegger, Igor Poltavsky, Kristof T. Schütt, Alexandre Tkatchenko, and Klaus-Robert Müller. Machine Learning Force Fields. *Chemical Reviews*, 121(16):10142–10186, August 2021b. ISSN 0009-2665, 1520-6890. doi: 10.1021/acs.chemrev.0c01111.

Chris G. Van de Walle and Jörg Neugebauer. First-principles calculations for defects and impurities: Applications to III-nitrides. *Journal of Applied Physics*, 95(8):3851–3879, April 2004. ISSN 0021-8979. doi: 10.1063/1.1682673.

Michiel J. van Setten, Fabio Caruso, Sahar Sharifzadeh, Xinguo Ren, Matthias Scheffler, Fang Liu, Johannes Lischner, Lin Lin, Jack R. Deslippe, Steven G. Louie, Chao Yang, Florian Weigend, Jeffrey B. Neaton, Ferdinand Evers, and Patrick Rinke. GW100: Benchmarking G0W0 for molecular systems. *Journal of Chemical Theory and Computation*, 11(12):5665–5687, 2015. doi: 10.1021/acs.jctc.5b00453.

Lynn E. Vescelius and Arthur A. Frost. Ellipsoidal Gaussian molecular calculations. *The Journal of Chemical Physics*, 61(8):2983–2987, 1974.

Oleg A. Vydrov and Troy Van Voorhis. Nonlocal van der Waals density functional: The simpler the better. *The Journal of Chemical Physics*, 133(24):244103, 2010. doi: 10.1063/1.3521275.

Chiduru Watanabe, Hirofumi Watanabe, Yoshio Okiyama, Daisuke Takaya, Kaori Fukuzawa, Shigenori Tanaka, and Teruki Honma. Development of an automated fragment molecular orbital (FMO) calculation protocol toward construction of quantum mechanical calculation database for large biomolecules. *Chem-Bio Informatics Journal*, 19:5–18, 2019. doi: 10.1273/cbij.19.5.

Florian Weigend and Reinhart Ahlrichs. Balanced basis sets of split valence, triple zeta valence and quadruple zeta valence quality for H to Rn: Design and assessment of accuracy. *Physical Chemistry Chemical Physics*, 7(18):3297–3305, 2005. doi: 10.1039/b508541a.

Steven R. White. Hybrid grid/basis set discretizations of the Schrödinger equation. *The Journal of Chemical Physics*, 147(24):244102, 2017.

Matthias Zwicker, Hanspeter Pfister, Jeroen van Baar, and Markus Gross. Surface splatting. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, pp. 371–378, 2001.

Scaling Density Functional Theory with Gaussian Splatting (Supplementary Material)

A Theoretical properties of GS-DFT **18**

A.1 Symmetry of the representation 18

A.2 Nuclear forces 19

A.3 Adaptive density fitting 22

A.4 Regularized eigensolve 24

B From Gaussian-type orbitals to floating splats **26**

B.1 Contracted Gaussian bases 26

B.2 Gaussian-type orbitals as limits of splats 28

B.3 Floating-orbital models 30

B.4 Correspondence with 3D Gaussian Splatting 32

C Implementation **33**

C.1 Software and numerics 34

C.2 Integrals 34

C.3 Complexity and memory 36

C.4 Multi-device sharding 37

C.5 Training recipe 38

C.6 Evaluation protocol 39

C.7 Reproducibility 40

D Comparison with differentiable DFT frameworks **41**

D.1 Capabilities 41

D.2 Protocol 41

D.3 Agreement on a small molecule 42

D.4 Scaling with system size 42

E Ablations **42**

E.1 Regularized eigensolve 42

E.2 Optimizer 43

E.3 Initialization 43

E.4 Refresh period 44

E.5 Screening threshold 44

F Additional results **45**

F.1 Bare anions 45

F.2 Dissociation energy ladders 45

A THEORETICAL PROPERTIES OF GS-DFT

GS-DFT replaces a fixed, atom-centered basis with floating 3D Gaussians whose parameters the optimizer is free to move. The extra degrees of freedom are what make the representation expressive, trading off the theoretical guarantees that come for free with an atom-centered basis. This appendix restores them one at a time. Specifically, we provide the theoretical frame for the following questions central to DFT calculations:

- **Section A.1**: Does a representation that is no longer attached to the atoms still respect the symmetries of the molecule?
- **Section A.2**: Are the forces obtained by differentiating the energy correct, and how wrong are they when the optimization stops early?
- **Section A.3**: What does the adaptive density fit cost in accuracy, and can the optimizer exploit it?
- **Section A.4** What does the regularized eigensolve change about the final answer?

Each subsection explains why the question matters, introduces the objects needed to answer it, and closes with what the implications of its results.

A.1 SYMMETRY OF THE REPRESENTATION

The energy of a molecule does not depend on where it sits or how it is oriented. For an atom-centered basis, this invariance holds automatically because every basis function is attached to a nucleus and moves with it. A splat cloud is not attached to anything. Its centers, orientations, and scales are free parameters, so invariance is no longer automatic. Energy invariance holds only if every rigid motion of the molecule can be absorbed by a transformation of the parameters that leaves the energy unchanged. Proposition A.1 shows that such a transformation exists, is explicit, and acts on the parameterization of Equation (8) as a group action. The symmetry is, therefore, a property of the naturally equivariant model architecture, not something the optimizer has to learn.

Two facts about the parameterization are needed. First, unit quaternions $\boldsymbol{q} \in S^3$ form a group under the Hamilton product $\otimes$, and the map $\boldsymbol{U} : S^3 \to SO(3)$ that sends a quaternion to its rotation matrix is a surjective homomorphism,

$$\boldsymbol{U}(\boldsymbol{q}_1 \otimes \boldsymbol{q}_2) = \boldsymbol{U}(\boldsymbol{q}_1)\,\boldsymbol{U}(\boldsymbol{q}_2)\ ,$$

whose kernel is $\{\pm 1\}$. The sign of $\boldsymbol{q}$ is therefore a discrete gauge: $\boldsymbol{q}$ and $-\boldsymbol{q}$ define the same splat. Second, every $(\boldsymbol{q}, \boldsymbol{l}) \in S^3 \times \mathbb{R}^3$ yields a symmetric positive-definite precision matrix

$$\boldsymbol{A} = \boldsymbol{U}(\boldsymbol{q})\,\mathrm{diag}(e^{\boldsymbol{l}})\,\boldsymbol{U}(\boldsymbol{q})^\top\ ,$$

with the closed forms

$$\begin{aligned}\boldsymbol{A}^{-1} &= \boldsymbol{U}\mathrm{diag}(e^{-\boldsymbol{l}})\,\boldsymbol{U}^\top\ ,\\ \det \boldsymbol{A} &= e^{\sum_i \ell_i}\ ,\\ \lambda_{\min}(\boldsymbol{A}) &= e^{\min_i \ell_i}\ .\end{aligned}$$

The optimization is therefore unconstrained, and the normalization in Equation (8) and the scale of every splat are available without a matrix decomposition.

Proposition A.1 (Equivariance of the splat parameterization)**.** *Let $\phi(\boldsymbol{r}) = \boldsymbol{Y}\boldsymbol{r} + \boldsymbol{b}$ be a rigid motion with $\boldsymbol{Y} \in O(3)$ and $\boldsymbol{b} \in \mathbb{R}^3$. Choose a unit quaternion $\boldsymbol{q}_{\boldsymbol{Y}}$ with $\boldsymbol{U}(\boldsymbol{q}_{\boldsymbol{Y}}) = \boldsymbol{Y}$ if $\det \boldsymbol{Y} = 1$, and $\boldsymbol{U}(\boldsymbol{q}_{\boldsymbol{Y}}) = -\boldsymbol{Y}$ if $\det \boldsymbol{Y} = -1$. Map each splat as*

$$\theta_\mu = (\boldsymbol{m}_\mu,\ \boldsymbol{q}_\mu,\ \boldsymbol{l}_\mu)\ \longmapsto\ \theta'_\mu = (\boldsymbol{Y}\boldsymbol{m}_\mu + \boldsymbol{b},\ \boldsymbol{q}_{\boldsymbol{Y}} \otimes \boldsymbol{q}_\mu,\ \boldsymbol{l}_\mu),$$

keep the coefficients $\boldsymbol{C}$, and move the nuclei to $\phi(\boldsymbol{R}_a)$. Then, for every splat μ and every orbital k,

$$\begin{aligned} g_{\theta'_\mu} \circ \phi &= g_{\theta_\mu},\\ \psi'_k \circ \phi &= \psi_k,\\ \rho' \circ \phi &= \rho,\end{aligned}$$

and every term of Equation (3) *takes the same value, as does the density-fitted Hartree energy of Equation* (14)*.*

Proof. The homomorphism property gives

$$\begin{aligned} \boldsymbol{U}(\boldsymbol{q}_{\boldsymbol{Y}} \otimes \boldsymbol{q}_\mu) &= \boldsymbol{Y}\,\boldsymbol{U}(\boldsymbol{q}_\mu) && \text{if } \det \boldsymbol{Y} = 1, \\ \boldsymbol{U}(\boldsymbol{q}_{\boldsymbol{Y}} \otimes \boldsymbol{q}_\mu) &= -\boldsymbol{Y}\,\boldsymbol{U}(\boldsymbol{q}_\mu) && \text{if } \det \boldsymbol{Y} = -1. \end{aligned}$$

In both cases the sign cancels in the quadratic form, so

$$\begin{aligned} \boldsymbol{A}'_\mu &= \boldsymbol{Y}\,\boldsymbol{U}(\boldsymbol{q}_\mu)\,\mathrm{diag}(e^{\boldsymbol{l}_\mu})\,\boldsymbol{U}(\boldsymbol{q}_\mu)^\top \boldsymbol{Y}^\top \\ &= \boldsymbol{Y}\,\boldsymbol{A}_\mu \boldsymbol{Y}^\top, \end{aligned}$$

and $\det \boldsymbol{A}'_\mu = \det \boldsymbol{A}_\mu$ because $(\det \boldsymbol{Y})^2 = 1$. The normalization of Equation (8) is unchanged, and for every $\boldsymbol{r}$,

$$\begin{aligned} (\phi(\boldsymbol{r}) - \boldsymbol{m}'_\mu)^\top \boldsymbol{A}'_\mu\,(\phi(\boldsymbol{r}) - \boldsymbol{m}'_\mu) &= (\boldsymbol{r} - \boldsymbol{m}_\mu)^\top \boldsymbol{Y}^\top \boldsymbol{Y}\,\boldsymbol{A}_\mu \boldsymbol{Y}^\top \boldsymbol{Y}\,(\boldsymbol{r} - \boldsymbol{m}_\mu) \\ &= (\boldsymbol{r} - \boldsymbol{m}_\mu)^\top \boldsymbol{A}_\mu\,(\boldsymbol{r} - \boldsymbol{m}_\mu), \end{aligned}$$

so $g_{\theta'_\mu} \circ \phi = g_{\theta_\mu}$. Because $|\det \boldsymbol{Y}| = 1$, the change of variables $\boldsymbol{r} \mapsto \phi(\boldsymbol{r})$ preserves every integral, so the overlap matrix $\boldsymbol{S}$ of Equation (9) is unchanged. The occupied Gram matrix and the Löwdin coefficients $\boldsymbol{C}$ of Equation (11) are then unchanged, and so is the density matrix $\boldsymbol{P}$. It follows that $\psi'_k \circ \phi = \psi_k$ and $\rho' \circ \phi = \rho$.

Each energy term is invariant for a separate reason. For the kinetic energy,

$$\boldsymbol{\nabla}\big(\psi_k \circ \phi^{-1}\big) = \boldsymbol{Y}\,(\boldsymbol{\nabla}\psi_k) \circ \phi^{-1},$$

and $\boldsymbol{Y}$ preserves norms. For the external energy,

$$|\phi(\boldsymbol{r}) - \phi(\boldsymbol{R}_a)| = |\boldsymbol{r} - \boldsymbol{R}_a|,$$

so $v'_{\mathrm{ext}} \circ \phi = v_{\mathrm{ext}}$. The Hartree and background terms depend only on the distances $|\boldsymbol{r} - \boldsymbol{r}'|$ and $|\boldsymbol{R}_a - \boldsymbol{R}_b|$, which ϕ preserves. Semilocal exchange-correlation functionals depend on ρ and on the rotational scalars $|\boldsymbol{\nabla}\rho|^2$, $\nabla^2\rho$, and the kinetic-energy density $\frac{1}{2}\sum_k |\boldsymbol{\nabla}\psi_k|^2$. Exact exchange (the Fock term in chemistry) is a Coulomb integral of orbital pair densities, so both are invariant. Finally, each auxiliary function of the adaptive fit is a pair product $g_\mu g_\nu$, so it transforms like the splats, and the matrices $\boldsymbol{Q}$ and $\boldsymbol{t}$ of Equation (14) are unchanged. □

The proposition holds for the full orthogonal group $O(3)$. A reflection needs no special treatment because a precision matrix is quadratic in the frame, so $\boldsymbol{Y}$ and $-\boldsymbol{Y}$ act identically. The quaternion never has to represent an improper rotation. The log-eigenvalues $\boldsymbol{l}$ are frame invariants, so the scales of a cloud carry no information about its orientation. Only the centers and the orientations transform, the latter by left multiplication:

$$\begin{aligned} \boldsymbol{m} &\mapsto \boldsymbol{Y}\boldsymbol{m} + \boldsymbol{b}, \\ \boldsymbol{q} &\mapsto \boldsymbol{q}_{\boldsymbol{Y}} \otimes \boldsymbol{q}. \end{aligned}$$

The result thus concerns the objective and not the optimizer. A converged cloud for one geometry maps to a converged cloud for every rigidly moved copy with the same energy, so no re-optimization is needed. Two rotated copies optimized from scratch will, in general, follow different trajectories because Adam's per-coordinate preconditioning is not rotation equivariant.

A.2 Nuclear forces

Nuclear forces are the negative gradient of the potential-energy surface

$$E^\star(\boldsymbol{R}) = \min_\theta\, E(\boldsymbol{R}, \theta),$$

the ground-state energy as a function of the nuclear coordinates. They drive geometry optimization and molecular dynamics, and they are the labels on which machine-learned interatomic potentials are trained (Section 5.1). Because $E^\star$ is the value function of an inner optimization problem, computing its gradient is an instance of the bilevel differentiation familiar from hyperparameter optimization and implicit layers. At an exact minimizer, the envelope theorem gives the answer. The response of

the parameters to the nuclei does not contribute, because the energy is stationary in them, and only the explicit dependence of E on $\boldsymbol{R}$ remains.

For an atom-centered basis, this explicit dependence is expensive. Each basis function $\chi_\mu(\boldsymbol{r} - \boldsymbol{R}_{a(\mu)})$ moves with its nucleus, so $\boldsymbol{R}$ enters every integral of Equation (3) and not only the electron–nucleus attraction. The envelope theorem removes the response of the coefficients $\boldsymbol{C}$, but the derivative of the basis functions themselves remains. This is the Pulay force (Pulay, 1969). It requires the nuclear derivative of every one- and two-electron integral, and it does not vanish as the optimization converges, because a fixed basis is not variationally optimal with respect to its own position. A floating basis removes it (Hurley, 1954). The splat centers are optimized, so the motion of the basis is part of the inner problem and the envelope theorem removes it together with $\boldsymbol{C}$. What remains is the electrostatic force of Feynman (1939).

Two complications keep the envelope argument from applying verbatim. First, the optimizer stops at a finite residual, so we need to know how the force error depends on it. Second, the minimizer of $E(\boldsymbol{R}, \cdot)$ is never isolated. The energy depends on $\boldsymbol{C}$ only through the density matrix, and

$$\bar{\boldsymbol{C}}\bar{\boldsymbol{C}}^\top = \boldsymbol{C}\,(\boldsymbol{C}^\top \boldsymbol{S}\boldsymbol{C})^{-1}\boldsymbol{C}^\top$$

is unchanged under $\boldsymbol{C} \mapsto \boldsymbol{C}\boldsymbol{B}$ for every invertible $\boldsymbol{B} \in \mathbb{R}^{N_{\mathrm{occ}} \times N_{\mathrm{occ}}}$. The quaternion sign of Section A.1 and relabelings of the splats are further, discrete symmetries. The Hessian $\boldsymbol{H} = \partial_\theta^2 E$ is therefore singular at every minimizer, and the implicit function theorem cannot be applied to θ directly. We work modulo these symmetries.

Assumption A.2 (Nondegenerate minimum up to symmetry)**.** Near a minimizer of $E(\boldsymbol{R}_0, \cdot)$, the energy is three times continuously differentiable, and for every $\boldsymbol{R}$ near $\boldsymbol{R}_0$ the minimizers of $E(\boldsymbol{R}, \cdot)$ form a smooth manifold $\mathcal{M}_{\boldsymbol{R}}$. At every $\theta^\star \in \mathcal{M}_{\boldsymbol{R}}$, the Hessian $\boldsymbol{H}$ is positive definite on the normal space

$$N_{\theta^\star} = (T_{\theta^\star}\mathcal{M}_{\boldsymbol{R}})^\perp,$$

with smallest eigenvalue at least $\kappa > 0$.

The manifold $\mathcal{M}_{\boldsymbol{R}}$ contains the symmetry orbit of any one minimizer, together with any other flat directions, such as the parameters of a splat that carries no weight. We write $\boldsymbol{H}^+$ for the pseudo-inverse of $\boldsymbol{H}$, which inverts it on $N_{\theta^\star}$, and $\boldsymbol{J} = \partial_{\boldsymbol{R}}\partial_\theta E$ for the mixed second derivative.

Proposition A.3 (Floating-basis forces and their residual error)**.** *Let Assumption A.2 hold.*

(i) The surface $E^\star$ is differentiable, and at any $\theta^\star \in \mathcal{M}_{\boldsymbol{R}}$ the force is the explicit partial derivative, with no Pulay term:

$$\begin{aligned}\boldsymbol{F}_a^\star &= -\frac{\mathrm{d}E^\star}{\mathrm{d}\boldsymbol{R}_a} \\ &= -\left.\frac{\partial E}{\partial \boldsymbol{R}_a}\right|_{\theta^\star} \\ &= Z_a \int \mathrm{d}^3\boldsymbol{r}\ \rho^\star(\boldsymbol{r})\,\frac{\boldsymbol{r} - \boldsymbol{R}_a}{|\boldsymbol{r} - \boldsymbol{R}_a|^3} + \sum_{b \neq a} Z_a Z_b\,\frac{\boldsymbol{R}_a - \boldsymbol{R}_b}{|\boldsymbol{R}_a - \boldsymbol{R}_b|^3}.\end{aligned} \tag{15}$$

(ii) Let $\tilde{\theta}$ be close to $\mathcal{M}_{\boldsymbol{R}}$, and define its residual and computed force as

$$\boldsymbol{\varepsilon} = \partial_\theta E(\boldsymbol{R}, \tilde{\theta}),$$
$$\tilde{\boldsymbol{F}}_a = -\left.\frac{\partial E}{\partial \boldsymbol{R}_a}\right|_{\tilde{\theta}}.$$

Then

$$\tilde{\boldsymbol{F}} - \boldsymbol{F}^\star = (\partial_{\boldsymbol{R}}\theta^\star)^\top \boldsymbol{\varepsilon} + \mathcal{O}(\|\boldsymbol{\varepsilon}\|^2), \tag{16}$$
$$E(\boldsymbol{R}, \tilde{\theta}) - E^\star(\boldsymbol{R}) = \tfrac{1}{2}\,\boldsymbol{\varepsilon}^\top \boldsymbol{H}^+ \boldsymbol{\varepsilon} + \mathcal{O}(\|\boldsymbol{\varepsilon}\|^3), \tag{17}$$

where

$$\partial_{\boldsymbol{R}}\theta^\star = -\boldsymbol{H}^+ \boldsymbol{J}^\top$$

is the response of the minimizer normal to $\mathcal{M}_{\boldsymbol{R}}$. In particular,

$$\|\tilde{\boldsymbol{F}} - \boldsymbol{F}^\star\| \leq \kappa^{-1}\|\boldsymbol{J}\|\,\|\boldsymbol{\varepsilon}\| + \mathcal{O}(\|\boldsymbol{\varepsilon}\|^2).$$

(iii) For every $\tilde{\theta}$, not only near a minimizer, the net computed force equals the summed center components of the residual,

$$\sum_a \tilde{\boldsymbol{F}}_a = \sum_\mu \frac{\partial E}{\partial \boldsymbol{m}_\mu}\bigg|_{\tilde{\theta}}. \tag{18}$$

Proof. (i) Fix $\theta^\star \in \mathcal{M}_{\boldsymbol{R}}$ and restrict θ to the affine slice $\theta^\star + N_{\theta^\star}$. On the slice the Hessian is invertible, so the implicit function theorem applied to the normal component of $\partial_\theta E(\boldsymbol{R}, \theta) = 0$ gives a continuously differentiable map $\boldsymbol{R} \mapsto \theta^\star(\boldsymbol{R}) \in \mathcal{M}_{\boldsymbol{R}}$ through $\theta^\star$. By the chain rule,

$$\begin{aligned}\frac{\mathrm{d}E^\star}{\mathrm{d}\boldsymbol{R}} &= \partial_{\boldsymbol{R}} E + (\partial_{\boldsymbol{R}}\theta^\star)^\top \partial_\theta E \\ &= \partial_{\boldsymbol{R}} E,\end{aligned}$$

since $\partial_\theta E = 0$ on $\mathcal{M}_{\boldsymbol{R}}$. Every point of $\mathcal{M}_{\boldsymbol{R}}$ lies on such a map and $\mathrm{d}E^\star/\mathrm{d}\boldsymbol{R}$ is unique, so $\partial_{\boldsymbol{R}} E$ takes the same value on all of $\mathcal{M}_{\boldsymbol{R}}$. The splats float free of the nuclei, so the kinetic, Hartree and exchange–correlation terms depend on $\boldsymbol{R}$ only through θ, and so does the auxiliary basis of Equation (14). The explicit dependence is confined to E_{BG} and to E_{ext}, through the potential

$$v_{\mathrm{ext}}(\boldsymbol{r}) = -\sum_b \frac{Z_b}{|\boldsymbol{r} - \boldsymbol{R}_b|}.$$

Differentiating these two terms gives the last line of Equation (15).

(ii) Let $\theta^\star$ be the point of $\mathcal{M}_{\boldsymbol{R}}$ closest to $\tilde{\theta}$, so that $\boldsymbol{\delta} = \tilde{\theta} - \theta^\star \in N_{\theta^\star}$. Differentiating $\partial_\theta E = 0$ along $\mathcal{M}_{\boldsymbol{R}}$ shows $T_{\theta^\star}\mathcal{M}_{\boldsymbol{R}} \subseteq \ker \boldsymbol{H}$. Since $\boldsymbol{H}$ is symmetric, its range lies in $N_{\theta^\star}$, where it is invertible with $\|\boldsymbol{H}^+\| \le \kappa^{-1}$. Taylor expansion of the gradient gives

$$\begin{aligned}\boldsymbol{\varepsilon} &= \boldsymbol{H}\boldsymbol{\delta} + \mathcal{O}(\|\boldsymbol{\delta}\|^2), \\ \boldsymbol{\delta} &= \boldsymbol{H}^+\boldsymbol{\varepsilon} + \mathcal{O}(\|\boldsymbol{\varepsilon}\|^2).\end{aligned}$$

Expanding the explicit partial in the same way,

$$\begin{aligned}\partial_{\boldsymbol{R}} E(\boldsymbol{R}, \tilde{\theta}) &= \partial_{\boldsymbol{R}} E(\boldsymbol{R}, \theta^\star) + \boldsymbol{J}\boldsymbol{\delta} + \mathcal{O}(\|\boldsymbol{\delta}\|^2), \\ \tilde{\boldsymbol{F}} - \boldsymbol{F}^\star &= -\boldsymbol{J}\boldsymbol{H}^+\boldsymbol{\varepsilon} + \mathcal{O}(\|\boldsymbol{\varepsilon}\|^2).\end{aligned}$$

Differentiating $\partial_\theta E(\boldsymbol{R}, \theta^\star(\boldsymbol{R})) = 0$ in $\boldsymbol{R}$ gives

$$\boldsymbol{J}^\top + \boldsymbol{H}\, \partial_{\boldsymbol{R}}\theta^\star = 0,$$

whose solution in $N_{\theta^\star}$ is $\partial_{\boldsymbol{R}}\theta^\star = -\boldsymbol{H}^+\boldsymbol{J}^\top$. Its transpose is $-\boldsymbol{J}\boldsymbol{H}^+$, which proves Equation (16). For the energy,

$$\begin{aligned}E(\boldsymbol{R}, \tilde{\theta}) - E(\boldsymbol{R}, \theta^\star) &= \tfrac{1}{2}\,\boldsymbol{\delta}^\top \boldsymbol{H}\boldsymbol{\delta} + \mathcal{O}(\|\boldsymbol{\delta}\|^3) \\ &= \tfrac{1}{2}\,\boldsymbol{\varepsilon}^\top \boldsymbol{H}^+\boldsymbol{\varepsilon} + \mathcal{O}(\|\boldsymbol{\varepsilon}\|^3),\end{aligned}$$

which is Equation (17).

(iii) By Proposition A.1 with $\boldsymbol{Y} = \mathbb{1}$, translating every nucleus and every splat center by the same $\boldsymbol{b}$ leaves E unchanged, for any $\tilde{\theta}$:

$$E\big(\{\boldsymbol{R}_a + \boldsymbol{b}\},\, \{\boldsymbol{m}_\mu + \boldsymbol{b}\},\, \dots\big) = E(\boldsymbol{R}, \tilde{\theta}).$$

Differentiating at $\boldsymbol{b} = 0$ gives

$$\sum_a \frac{\partial E}{\partial \boldsymbol{R}_a} + \sum_\mu \frac{\partial E}{\partial \boldsymbol{m}_\mu} = 0,$$

which is Equation (18). □

The proposition treats the auxiliary basis as a function of the cloud, which it is at every refresh. Between refreshes, the optimizer follows the gradient with the auxiliary basis held fixed (Algorithm 1). That gradient omits the dependence of the fit on the auxiliary functions, and the omitted term is proportional to the Coulomb-norm fit residual that Section A.3 controls.

Proposition A.3 allows three statements in the main text. First, a force costs one reverse-mode pass through two terms of the energy, with no derivative integrals and no Pulay correction. Second, the energy error is quadratic in the residual and the force error is linear. Along one optimization, forces therefore converge more slowly than energies even at a fixed cloud size, so force labels need a tighter stopping criterion than energies do, and the gradient norm that the optimizer already reports bounds their error. Third, Equation (18) is a free and reference-free check. An exact force field has zero net force, and the computed net force is exactly the summed center gradient, so a nonzero net force measures incomplete optimization and nothing else.

A.3 Adaptive density fitting

The Hartree energy is a six-dimensional integral that is quartic in the basis (Equation (6)), and density fitting replaces it with a quadratic one by expanding the density in an auxiliary basis. In GS-DFT, the fitted energy sits inside the objective that the optimizer minimizes. This raises two questions. The first is how accurate the fit is. The second is familiar from learning with a surrogate inside the loss: can the optimizer lower the objective by degrading the surrogate instead of improving the physics? Proposition A.4 answers the first for any auxiliary basis, and Proposition A.5 answers the second for the adaptive basis of Section 4.

The object that makes both answers exact is the Coulomb inner product,

$$\begin{aligned}(f \mid g) &= \int \mathrm{d}^3\boldsymbol{r} \int \mathrm{d}^3\boldsymbol{r}' \; \frac{f(\boldsymbol{r})\, g(\boldsymbol{r}')}{|\boldsymbol{r}-\boldsymbol{r}'|} \\ &= \int \frac{\mathrm{d}^3\boldsymbol{k}}{(2\pi)^3} \, \frac{4\pi}{|\boldsymbol{k}|^2} \, \overline{\hat{f}(\boldsymbol{k})}\, \hat{g}(\boldsymbol{k}),\end{aligned} \tag{19}$$

defined for real functions of finite Coulomb energy, with $\hat{f}$ the Fourier transform. The kernel $4\pi/|\boldsymbol{k}|^2$ is strictly positive, so $(\cdot|\cdot)$ is an inner product with norm $\|f\|_{\mathrm{C}} = (f|f)^{1/2}$. In this notation, the Hartree energy of Equation (3) and the quantities of Equation (14) read

$$\begin{aligned}E_H[\rho] &= \tfrac{1}{2}\,\|\rho\|_{\mathrm{C}}^2, \\ \boldsymbol{Q}_{ab} &= (h_a \mid h_b), \\ \boldsymbol{t}_a &= (\rho \mid h_a).\end{aligned}$$

$\boldsymbol{Q}$ is the Gram matrix of the auxiliary functions, and it is positive definite whenever they are linearly independent. The density fit of Section 2 is therefore least squares in the Coulomb norm, and its solution defines the Coulomb-orthogonal projection of ρ onto $\mathrm{span}\{h_a\}$,

$$\begin{aligned}\boldsymbol{d} &= \boldsymbol{Q}^{-1}\boldsymbol{t}, \\ \Pi\rho &= \textstyle\sum_a d_a h_a.\end{aligned}$$

Proposition A.4 (Robust density fitting (Dunlap, 2000))**.** *Let $\{h_a\}_{a=1}^{N_{\mathrm{aux}}}$ be linearly independent functions of finite Coulomb energy, and let $\rho_{\boldsymbol{d}} = \sum_a d_a h_a$ for $\boldsymbol{d} \in \mathbb{R}^{N_{\mathrm{aux}}}$. For every $\boldsymbol{d}$,*

$$\boldsymbol{t}^\top \boldsymbol{d} - \tfrac{1}{2}\,\boldsymbol{d}^\top \boldsymbol{Q}\,\boldsymbol{d} = E_H[\rho] - \tfrac{1}{2}\,\|\rho - \rho_{\boldsymbol{d}}\|_{\mathrm{C}}^2. \tag{20}$$

The left-hand side is therefore a lower bound on $E_H[\rho]$, with an error that is quadratic in the fit error. It is maximized at $\boldsymbol{d} = \boldsymbol{Q}^{-1}\boldsymbol{t}$, where it equals

$$\tilde{E}_H = \tfrac{1}{2}\,\boldsymbol{t}^\top \boldsymbol{Q}^{-1}\boldsymbol{t},$$

and the gap to the exact Hartree energy is

$$E_H[\rho] - \tilde{E}_H = \tfrac{1}{2}\,\|\rho - \Pi\rho\|_{\mathrm{C}}^2 \;\geq\; 0. \tag{21}$$

For $\lambda \geq 0$, the regularized estimate satisfies

$$\begin{aligned}\tilde{E}_H^{\lambda} &= \tfrac{1}{2}\,\boldsymbol{t}^\top(\boldsymbol{Q} + \lambda\mathbb{1})^{-1}\boldsymbol{t}, \\ 0 \;\leq\; \tilde{E}_H - \tilde{E}_H^{\lambda} &\leq \frac{\lambda}{\lambda_{\min}(\boldsymbol{Q}) + \lambda}\,\tilde{E}_H.\end{aligned}$$

Proof. Expanding the square,

$$\begin{aligned}\tfrac{1}{2}\|\rho - \rho_{\boldsymbol{d}}\|_{\mathrm{C}}^2 &= \tfrac{1}{2}(\rho|\rho) - \sum_a d_a(\rho|h_a) + \tfrac{1}{2}\sum_{ab} d_a d_b(h_a|h_b)\\ &= E_H[\rho] - \boldsymbol{t}^\top \boldsymbol{d} + \tfrac{1}{2}\,\boldsymbol{d}^\top \boldsymbol{Q}\boldsymbol{d},\end{aligned}$$

which rearranges to Equation (20). The left-hand side is strictly concave in $\boldsymbol{d}$, with stationary point $\boldsymbol{Q}\boldsymbol{d} = \boldsymbol{t}$. Substituting $\boldsymbol{d} = \boldsymbol{Q}^{-1}\boldsymbol{t}$ gives $\tilde{E}_H$, and the right-hand side of Equation (20) at this point gives Equation (21). For the regularized estimate,

$$\boldsymbol{Q}^{-1} - (\boldsymbol{Q} + \lambda\mathbb{1})^{-1} = \lambda\,\boldsymbol{Q}^{-1}(\boldsymbol{Q} + \lambda\mathbb{1})^{-1}.$$

In the eigenbasis of $\boldsymbol{Q}$, with eigenvalues q_i, the right-hand side is diagonal with entries $q_i^{-1}\,\lambda/(q_i+\lambda)$, so

$$0 \;\preceq\; \boldsymbol{Q}^{-1} - (\boldsymbol{Q} + \lambda\mathbb{1})^{-1} \;\preceq\; \frac{\lambda}{\lambda_{\min}(\boldsymbol{Q}) + \lambda}\,\boldsymbol{Q}^{-1}.$$

Taking the quadratic form in $\boldsymbol{t}$ gives the bound. □

Equation (20) is why the fit is called robust. A first-order error in the fitted coefficients produces only a second-order error in the energy, because the energy is stationary in $\boldsymbol{d}$ at the fit. Equation (21) makes the error a geometric quantity: half the squared Coulomb distance from ρ to the auxiliary span. The regularizer $\lambda = 10^{-8}$ used in all experiments keeps the Cholesky factorization of Algorithm 1 stable when two auxiliary functions nearly coincide, and it can only lower the estimate further.

For the adaptive basis, the span is built from the density itself. By the Gaussian product theorem, the product of two splats is a single Gaussian,

$$g_\mu(\boldsymbol{r})\, g_\nu(\boldsymbol{r}) \propto h_{\mu\nu}(\boldsymbol{r}), \tag{22}$$

with the proportionality constant of Equation (9), where $h_{\mu\nu}$ has precision and center

$$\begin{aligned}\boldsymbol{A}_{\mu\nu} &= \boldsymbol{A}_\mu + \boldsymbol{A}_\nu,\\ \boldsymbol{m}_{\mu\nu} &= (\boldsymbol{A}_\mu + \boldsymbol{A}_\nu)^{-1}(\boldsymbol{A}_\mu\boldsymbol{m}_\mu + \boldsymbol{A}_\nu\boldsymbol{m}_\nu).\end{aligned}$$

The density $\rho = \sum_{\mu\nu} \boldsymbol{P}_{\mu\nu}\, g_\mu g_\nu$ of Equation (5) is a linear combination of these products. Screening retains a set of unordered pairs and splits the density into a retained and a discarded part,

$$\begin{aligned}\mathcal{P} &= \big\{\{\mu,\nu\} : |\boldsymbol{S}_{\mu\nu}| > \tau\big\},\\ \rho &= \rho_{\mathcal{P}} + \rho_{\mathcal{P}^c}.\end{aligned}$$

The auxiliary basis is refreshed every K steps and held fixed in between (Algorithm 1). We write Π_r for the projection onto the span built from the cloud Θ_r at the most recent refresh, and ρ_r for the density at that refresh.

Proposition A.5 (Closure of the adaptive fit)**.** *At a refresh, the fit error is confined to the discarded pairs,*

$$\begin{aligned}E_H[\rho_r] - \tilde{E}_H &= \tfrac{1}{2}\,\|(1-\Pi_r)\,\rho_{r,\mathcal{P}^c}\|_{\mathrm{C}}^2\\ &\le \tfrac{1}{2}\,\|\rho_{r,\mathcal{P}^c}\|_{\mathrm{C}}^2,\end{aligned} \tag{23}$$

and the fit is exact for $\tau = 0$. At any later step before the next refresh, with density ρ,

$$\Big(E_H[\rho] - \tilde{E}_H\Big)^{1/2} \;\le\; \tfrac{1}{\sqrt{2}}\Big(\|\rho_{r,\mathcal{P}^c}\|_{\mathrm{C}} + \|\rho - \rho_r\|_{\mathrm{C}}\Big). \tag{24}$$

The same statements hold for each exchange pair density $\rho_{kl} = \bar{\psi}_k\bar{\psi}_l$. With $(\boldsymbol{t}_{kl})_a = (\rho_{kl}|h_a)$, the fitted and exact exchange energies are

$$\begin{aligned}\tilde{E}_x &= -\sum_{kl} \boldsymbol{t}_{kl}^\top \boldsymbol{Q}^{-1}\boldsymbol{t}_{kl},\\ E_x &= -\sum_{kl} \|\rho_{kl}\|_{\mathrm{C}}^2,\end{aligned}$$

and the fit is an upper bound, $\tilde{E}_x \ge E_x$.

Proof. Every retained product is proportional to an auxiliary function, so $\rho_{r,\mathcal{P}} \in \operatorname{span}\{h_a\}$ and

$$\Pi_r \rho_{r,\mathcal{P}} = \rho_{r,\mathcal{P}},$$
$$\rho_r - \Pi_r \rho_r = (1 - \Pi_r)\, \rho_{r,\mathcal{P}^c}.$$

Equation (21) then gives the identity in Equation (23). The inequality holds because an orthogonal projection does not increase the norm, and for $\tau = 0$ the discarded part is empty. For Equation (24), write $\rho = \rho_r + (\rho - \rho_r)$ and use the triangle inequality together with $\|(1 - \Pi_r) f\|_{\mathrm{C}} \le \|f\|_{\mathrm{C}}$,

$$\begin{aligned}\|(1 - \Pi_r)\, \rho\|_{\mathrm{C}} &\le \|(1 - \Pi_r)\, \rho_r\|_{\mathrm{C}} + \|(1 - \Pi_r)(\rho - \rho_r)\|_{\mathrm{C}} \\ &\le \|\rho_{r,\mathcal{P}^c}\|_{\mathrm{C}} + \|\rho - \rho_r\|_{\mathrm{C}},\end{aligned}$$

and apply Equation (21). For exchange, the Löwdin orbitals are linear combinations of splats, so each ρ_{kl} is again a linear combination of splat products and the same argument applies. Applying Equation (21) to each ρ_{kl} gives

$$\|\rho_{kl}\|_{\mathrm{C}}^2 \ge \boldsymbol{t}_{kl}^\top \boldsymbol{Q}^{-1} \boldsymbol{t}_{kl},$$

and the minus sign in E_x reverses the inequality. □

The answer to the second question follows. The objective the optimizer minimizes is the exact-integral energy minus the gap of Equation (21), so a step that enlarges the gap lowers the objective without improving the physics. The fit coefficients are solved in closed form, so there is no inner adversary to train and no alternating min–max dynamics. The only exploitable quantity is the gap itself, and Proposition A.5 bounds it. At every refresh it resets to the contribution of the discarded pairs, which carry charge $|\boldsymbol{P}_{\mu\nu} \boldsymbol{S}_{\mu\nu}| \le \tau |\boldsymbol{P}_{\mu\nu}|$ each. Between refreshes it can grow only with the distance the density has moved since the last refresh. This is the trade-off that the refresh period K controls: a larger K amortizes the cubic factorization of Algorithm 1 over more steps, and allows the density to drift further from the span it was fitted in.

Two consequences for the reported energies follow. For semilocal functionals only the Hartree term is fitted, so the fitted energy lies below the exact-integral energy at the same parameters, by the gap. For hybrid functionals and Hartree–Fock, the Hartree and exchange errors have opposite signs, and the fitted energy is not a bound in either direction. In both cases the error vanishes as $\tau \to 0$ and $K \to 1$.

A.4 Regularized eigensolve

The Löwdin retraction of Equation (11) needs the inverse square root of the occupied Gram matrix

$$\boldsymbol{G} = \boldsymbol{C}^\top \boldsymbol{S} \boldsymbol{C},$$

and both its value and its gradient can fail. In the forward pass, $\boldsymbol{G}^{-1/2}$ diverges as the smallest eigenvalue of $\boldsymbol{G}$ goes to zero. In a floating basis this is not an edge case. The energy is invariant under $\boldsymbol{C} \mapsto \boldsymbol{C}\boldsymbol{B}$ (Section A.2), so nothing in the objective keeps the columns of $\boldsymbol{C}$ from becoming nearly parallel, and splats that drift on top of each other make $\boldsymbol{S}$ itself nearly singular. In the backward pass, the standard gradient of an eigendecomposition contains the factors $(\lambda_i - \lambda_j)^{-1}$, which diverge at degenerate eigenvalues even when the forward pass is harmless. This is not an edge case either: an orthonormal initialization gives $\boldsymbol{G} = \mathbb{1}$, where every pair of eigenvalues is degenerate. The same failure appears in machine learning wherever a network differentiates through an eigendecomposition, as in whitening layers. Proposition A.6 states what the regularization of Equation (12) changes, and shows that the divergence of the backward pass is an artifact of how the gradient is assembled rather than a property of the derivative.

What we need is the derivative of a matrix function. For a symmetric $\boldsymbol{G} = \boldsymbol{U} \operatorname{diag}(\boldsymbol{\lambda})\, \boldsymbol{U}^\top$ and a scalar function f differentiable on the spectrum, the matrix function $f(\boldsymbol{G}) = \boldsymbol{U} \operatorname{diag}(f(\boldsymbol{\lambda}))\, \boldsymbol{U}^\top$ has the Fréchet derivative (Bhatia, 1997, Thm. V.3.3)

$$\mathrm{D} f(\boldsymbol{G})[\dot{\boldsymbol{G}}] = \boldsymbol{U} \big(\boldsymbol{\Omega} \circ \boldsymbol{U}^\top \dot{\boldsymbol{G}}\, \boldsymbol{U} \big)\, \boldsymbol{U}^\top, \tag{25}$$

where $\circ$ is the elementwise product and the kernel is the first divided difference $f^{[1]}$ of f,

$$\Omega_{ij} = f^{[1]}(\lambda_i, \lambda_j),$$
$$f^{[1]}(\lambda_i, \lambda_j) = \begin{cases} \dfrac{f(\lambda_i) - f(\lambda_j)}{\lambda_i - \lambda_j}, & \lambda_i \ne \lambda_j, \\ f'(\lambda_i), & \lambda_i = \lambda_j. \end{cases}$$

By the mean value theorem, for some ξ between λ_i and λ_j,

$$f^{[1]}(\lambda_i, \lambda_j) = f'(\xi),$$

so $\boldsymbol{\Omega}$ is bounded wherever f' is. The derivative itself therefore has no singularity at degenerate eigenvalues. The standard eigendecomposition gradient differentiates $\boldsymbol{U}$ and $\boldsymbol{\lambda}$ separately, and each of those derivatives is singular at a degeneracy, even though their combination in Equation (25) is not.

The forward pass of Equation (12) uses

$$\begin{aligned} f(\lambda) &= \max(\lambda, \varphi)^{-1/2}, \\ \varphi &= \varepsilon \lambda_{\max}, \\ \boldsymbol{T} &= f(\boldsymbol{G}), \\ \bar{\boldsymbol{C}} &= \boldsymbol{C}\boldsymbol{T}, \end{aligned}$$

and we call an eigenvalue floored when $\lambda_i < \varphi$. The regularized backward pass replaces $\boldsymbol{\Omega}$ by

$$\tilde{\Omega}_{ij} = \begin{cases} 0, & \lambda_i < \varphi \text{ or } \lambda_j < \varphi, \\ \frac{1}{2}\big(f'(\lambda_i) + f'(\lambda_j)\big), & |\lambda_i - \lambda_j| \le \delta \lambda_{\max}, \\ \Omega_{ij}, & \text{otherwise.} \end{cases}$$

All derivatives treat the floor level φ as a constant.

Proposition A.6 (Regularized Löwdin retraction)**.** *Let $\boldsymbol{S}$ be positive semidefinite, so that $\boldsymbol{G} \succeq 0$, and let $\lambda_{\max} > 0$.*

(i) If $\lambda_{\min}(\boldsymbol{G}) \ge \varepsilon \lambda_{\max}$, that is, if $\operatorname{cond}(\boldsymbol{G}) \le \varepsilon^{-1}$, then exactly

$$\begin{aligned} \boldsymbol{T} &= \boldsymbol{G}^{-1/2}, \\ \bar{\boldsymbol{C}}^{\top} \boldsymbol{S} \bar{\boldsymbol{C}} &= \mathbb{1}. \end{aligned}$$

(ii) In general, $\|\boldsymbol{T}\| \le \varphi^{-1/2}$, and the electron count of $\boldsymbol{P} = 2\bar{\boldsymbol{C}}\bar{\boldsymbol{C}}^{\top}$ is

$$\begin{aligned} \operatorname{Tr}(\boldsymbol{P}\boldsymbol{S}) &= 2 \sum_{i=1}^{N_{\mathrm{occ}}} \min\Big(1, \frac{\lambda_i}{\varphi}\Big) \\ &\le 2N_{\mathrm{occ}}, \end{aligned} \tag{26}$$

with equality if and only if no eigenvalue is floored.

(iii) For every pair of unfloored eigenvalues, $\tilde{\Omega}_{ij} = \Omega_{ij}$ if $|\lambda_i - \lambda_j| > \delta \lambda_{\max}$, and otherwise

$$\big|\tilde{\Omega}_{ij} - \Omega_{ij}\big| \;\le\; \frac{5}{32} \frac{(\lambda_i - \lambda_j)^2}{\min(\lambda_i, \lambda_j)^{7/2}}. \tag{27}$$

To leading order, the relative error is

$$\frac{\big|\tilde{\Omega}_{ij} - \Omega_{ij}\big|}{|\Omega_{ij}|} \approx \frac{5}{16} \frac{(\lambda_i - \lambda_j)^2}{\lambda_i^2}.$$

(iv) The regularized gradient is the exact gradient with respect to perturbations of $\boldsymbol{G}$ restricted to the unfloored eigenspace, up to the error in (iii). If no eigenvalue is floored and no two eigenvalues lie within $\delta \lambda_{\max}$ of each other, it is the exact gradient.

Proof. (i) If no eigenvalue is floored, $f(\lambda) = \lambda^{-1/2}$ on the spectrum, so $\boldsymbol{T} = \boldsymbol{G}^{-1/2}$ and

$$\begin{aligned} \bar{\boldsymbol{C}}^{\top} \boldsymbol{S} \bar{\boldsymbol{C}} &= \boldsymbol{T}\, \boldsymbol{C}^{\top} \boldsymbol{S} \boldsymbol{C}\, \boldsymbol{T} \\ &= \boldsymbol{G}^{-1/2}\, \boldsymbol{G}\, \boldsymbol{G}^{-1/2} \\ &= \mathbb{1}. \end{aligned}$$

(ii) The eigenvalues of $\boldsymbol{T}$ are $\max(\lambda_i, \varphi)^{-1/2} \le \varphi^{-1/2}$. Since $\boldsymbol{T}$ and $\boldsymbol{G}$ share eigenvectors,

$$\begin{aligned}\bar{\boldsymbol{C}}^\top \boldsymbol{S} \bar{\boldsymbol{C}} &= \boldsymbol{T}\boldsymbol{G}\boldsymbol{T} \\ &= \boldsymbol{U}\mathrm{diag}\Big(\frac{\lambda_i}{\max(\lambda_i, \varphi)}\Big)\boldsymbol{U}^\top,\end{aligned}$$

and $\mathrm{Tr}(\boldsymbol{P}\boldsymbol{S}) = 2\,\mathrm{Tr}(\bar{\boldsymbol{C}}^\top \boldsymbol{S}\bar{\boldsymbol{C}})$ gives Equation (26). Each term is at most one, with equality exactly when $\lambda_i \ge \varphi$.

(iii) Above the floor, $f(\lambda) = \lambda^{-1/2}$. The divided difference is the average of f' over the interval between λ_j and λ_i, and $\tilde{\Omega}_{ij}$ is the trapezoidal estimate of that average. On an interval of length h, the trapezoidal rule integrates f' with error $h^3|f'''(\xi)|/12$ for some ξ in the interval, so the error in the average is $h^2|f'''(\xi)|/12$. The third derivative and the kernel are

$$\begin{aligned}|f'''(\lambda)| &= \tfrac{15}{8}\,\lambda^{-7/2}, \\ |\Omega_{ij}| \approx |f'(\lambda_i)| &= \tfrac{1}{2}\,\lambda_i^{-3/2}.\end{aligned}$$

The first is largest at the smaller eigenvalue, which gives Equation (27), and dividing by the second gives the relative error. On the diagonal, $\tilde{\Omega}_{ii} = f'(\lambda_i) = \Omega_{ii}$ exactly.

(iv) Let $\boldsymbol{U}_+$ collect the eigenvectors of the unfloored eigenvalues. Because $\tilde{\Omega}_{ij} = 0$ whenever i or j is floored, the regularized backward pass maps an upstream gradient $\bar{\boldsymbol{T}}$ to

$$\bar{\boldsymbol{T}} \;\longmapsto\; \boldsymbol{U}_+\big(\tilde{\boldsymbol{\Omega}}_{++} \circ \boldsymbol{U}_+^\top \bar{\boldsymbol{T}}\,\boldsymbol{U}_+\big)\,\boldsymbol{U}_+^\top.$$

By Equation (25), this is the adjoint of the restricted derivative

$$\dot{\boldsymbol{G}} \;\longmapsto\; \mathrm{D}f(\boldsymbol{G})\big[\boldsymbol{U}_+\boldsymbol{U}_+^\top \dot{\boldsymbol{G}}\,\boldsymbol{U}_+\boldsymbol{U}_+^\top\big],$$

with $\boldsymbol{\Omega}_{++}$ replaced by $\tilde{\boldsymbol{\Omega}}_{++}$. By (iii), the two agree when no two unfloored eigenvalues lie within $\delta\lambda_{\max}$, and if no eigenvalue is floored the projection is the identity. □

The proposition separates what the regularization does in the two passes. The forward floor is inactive on a well-conditioned Gram matrix, and when it engages it can only remove charge, never add it, by the amount in Equation (26). The backward regularization is exact on the diagonal and between floored eigenvalues, and within a near-degenerate pair it makes a second-order error that is small unless the pair sits close to the floor. The one deliberate change to the gradient is that the directions coupling floored and unfloored modes are dropped. This is a choice we made to stop the gradient from acting on the orbital combinations that the retraction is failing to normalize.

Two consequences follow. First, at convergence the regularization has no effect whenever the converged Gram matrix has condition number below ε^{-1} and no near-degenerate eigenvalues. The optimizer then solves the exact stationarity condition, and Proposition A.3 applies unchanged. Second, whether this holds is checkable after the fact from the logged ratio $\lambda_{\min}(\boldsymbol{G})/\lambda_{\max}(\boldsymbol{G})$, and any floor that remains active shows up as the electron deficit of Equation (26).

B From Gaussian-type orbitals to floating splats

Quantum chemistry has refined fixed Gaussian basis sets for seventy years, and computer graphics has recently shown that a cloud of free anisotropic Gaussians, optimized end to end, can represent a complicated three-dimensional signal. This appendix makes the connection precise in both directions. Section B.1 defines a contracted Gaussian basis and counts its parameters, which is the accounting behind every comparison in Section 5. Section B.2 shows that splats contain Gaussian-type orbitals as limits, and what that inclusion costs. Section B.3 places the earlier floating-orbital models inside the GS-DFT family and identifies what stopped them. Section B.4 states the correspondence with 3DGS and proves where it stops.

B.1 Contracted Gaussian bases

Every Gaussian baseline in Section 5 belongs to a standard basis-set family, and every comparison there is made either at matched function count or per free parameter. Both comparisons require a precise definition of a basis function and of the numbers that define it.

The most basic building block is a primitive Gaussian pinned to a nucleus $\boldsymbol{R}_a$. Writing $\boldsymbol{s} = \boldsymbol{r} - \boldsymbol{R}_a$, a Cartesian primitive of angular momentum $l = n_x + n_y + n_z$ and exponent $\alpha > 0$ is

$$\chi^{\mathrm{prim}}(\boldsymbol{r}) = s_x^{n_x}\, s_y^{n_y}\, s_z^{n_z}\; e^{-\alpha|\boldsymbol{s}|^2}. \tag{28}$$

The $(l+1)(l+2)/2$ monomials of degree l are usually replaced by the $2l+1$ real solid harmonics

$$\mathcal{Y}_{lm}(\boldsymbol{s}) = |\boldsymbol{s}|^l\, Y_{lm}(\boldsymbol{s}/|\boldsymbol{s}|), \qquad m = -l, \dots, l,$$

which span the part of angular momentum l.

Definition B.1 (Contracted Gaussian basis)**.** A shell s on atom $a(s)$ consists of an angular momentum l_s, K_s exponents $\alpha_{sp} > 0$ and K_s contraction weights d_{sp}. It contributes the $2l_s + 1$ functions

$$\chi_{sm}(\boldsymbol{r}) = \mathcal{Y}_{l_s m}(\boldsymbol{r} - \boldsymbol{R}_{a(s)}) \sum_{p=1}^{K_s} d_{sp}\; e^{-\alpha_{sp}|\boldsymbol{r}-\boldsymbol{R}_{a(s)}|^2}, \qquad m = -l_s, \dots, l_s. \tag{29}$$

A basis set is a table that assigns a list of shells to every element. For a molecule it yields $B = \sum_s (2l_s + 1)$ functions, and the orbitals are expanded as

$$\psi_k(\boldsymbol{r}) = \sum_{\mu=1}^{B} \boldsymbol{C}_{\mu k}\, \chi_\mu(\boldsymbol{r}).$$

All $2l_s + 1$ functions of a shell share the same exponents and weights, so a shell is defined by $2K_s$ numbers, whatever its angular momentum. The exponents and weights are fitted once per element, largely to atomic calculations, and then stored (Boys, 1950; Ditchfield et al., 1971; Dunning Jr., 1989). During a calculation, only $\boldsymbol{C}$ is optimized.

Basis sets come in families indexed by a cardinal number ζ (Jensen, 2013). In the correlation-consistent family cc-pVζZ (Dunning Jr., 1989), with $\zeta = 2, 3, 4$ for D, T, Q, a first-row atom carries

$$n_l(\zeta) = \zeta + 1 - l, \qquad l = 0, \dots, \zeta,$$

contracted shells of angular momentum l, which gives $3s\,2p\,1d$, $4s\,3p\,2d\,1f$ and $5s\,4p\,3d\,2f\,1g$. Each step in ζ adds one shell of every angular momentum already present and raises the highest angular momentum by one. The number of functions per first-row atom is therefore

$$\begin{aligned} B_{\mathrm{atom}}(\zeta) &= \sum_{l=0}^{\zeta} (\zeta + 1 - l)(2l + 1) \\ &= \tfrac{1}{6}(\zeta + 1)(\zeta + 2)(2\zeta + 3), \end{aligned}$$

that is 14, 30 and 55, and hydrogen carries $B_{\mathrm{atom}}(\zeta - 1)$. The basis grows cubically in ζ. For self-consistent-field energies such as Hartree–Fock and DFT, the energy converges close to exponentially in ζ (Jensen, 2013),

$$E_\zeta \approx E_{\mathrm{CBS}} + A\, e^{-b\zeta},$$

and this regularity is what makes an extrapolation to the complete-basis-set (CBS) limit E_{CBS} meaningful. The def2 family (Weigend & Ahlrichs, 2005) uses segmented contractions designed for balanced accuracy across the periodic table, and we use its quadruple-zeta member def2-QZVPP where a large reference basis is needed.

A tabulated basis is often described as having no parameters, because nothing in it is optimized during a calculation. We count its tabulated numbers anyway. They are parameters of the representation in the machine-learning sense: values that had to be chosen, fitted to data, and shipped with the model for its outputs to mean anything, like the weights of a pretrained network that is used frozen. With the nine-parameter chart of Equation (8), the parameter counts of a basis of B functions and of a cloud of M splats are

$$P_{\mathrm{GTO}} = 2 \sum_s K_s + B\, N_{\mathrm{occ}}, \tag{30}$$

$$P_{\mathrm{splat}} = 9M + M\, N_{\mathrm{occ}}. \tag{31}$$

We write

$$\eta = \frac{2\sum_s K_s}{B}$$

for the tabulated numbers per Gaussian function. Across the cc-pVζZ bases used in this paper, $\eta \in [6.6, 7.5]$, against 9 for a splat, so both representations pay the same order per function. The basis part of each count scales with the number of functions and the coefficient part with the number of functions times N_{occ}, so

$$\frac{\text{basis parameters}}{\text{coefficient parameters}} = \frac{\eta}{N_{\mathrm{occ}}},$$

with $\eta = 9$ for splats. Table 4 evaluates this ratio for the three molecules of Section 5.

Table 4: Basis parameters per coefficient parameter, η/N_{occ}, with $\eta = 7$ for the Gaussian basis and $\eta = 9$ for splats.

	H_2O	C_2H_5OH	Ala dipeptide
N_{occ}	5	13	39
Gaussian basis	1.40	0.54	0.18
Splat cloud	1.80	0.69	0.23

At matched function count $M = B$, Equations (30) and (31) give the relative overhead of the cloud,

$$\frac{P_{\mathrm{splat}}}{P_{\mathrm{GTO}}} - 1 = \frac{9 - \eta}{\eta + N_{\mathrm{occ}}}, \tag{32}$$

which decreases monotonically in N_{occ}. For $\eta \geq 6.6$ it is at most 21% on water and at most 5.3% on alanine dipeptide. Water with cc-pVDZ is a concrete case, with $B = 24$ functions defined by 158 tabulated numbers:

$$\begin{aligned} P_{\mathrm{GTO}} &= 158 + 24 \cdot 5 \\ &= 278, \\ P_{\mathrm{splat}} &= 9 \cdot 24 + 24 \cdot 5 \\ &= 336. \end{aligned}$$

The ratio is 1.21, while counting only the coefficients on the Gaussian side would give 2.8. Matched function count is therefore also matched parameter count, up to the correction of Equation (32). This is why the accuracy-per-parameter comparison of Figure 5 counts the tabulated basis on the Gaussian side, and why a gain at matched function count is not bought with hidden parameters.

B.2 Gaussian-type orbitals as limits of splats

Every splat in Equation (8) is a positive function with no angular factor, while a Gaussian basis carries p, d and f functions whose angular factors create the nodal structure of an orbital. Section 4 claims that signed coefficients recover this structure from displaced splats of opposite sign. Proposition B.2 makes the claim precise, and Proposition B.3 shows what it costs.

We use isotropic Gaussians, which are splats with precision $\boldsymbol{A} = 2\alpha\mathbb{1}$ up to normalization,

$$e_{\boldsymbol{m}}(\boldsymbol{r}) = e^{-\alpha|\boldsymbol{r}-\boldsymbol{m}|^2}.$$

Two facts are needed. First, derivatives with respect to the center produce Hermite polynomials. For a multi-index $\boldsymbol{n} = (n_x, n_y, n_z)$ with $|\boldsymbol{n}| = n_x + n_y + n_z$,

$$\partial_{\boldsymbol{m}}^{\boldsymbol{n}}\, e_{\boldsymbol{m}}(\boldsymbol{r}) = \alpha^{|\boldsymbol{n}|/2} \prod_{i\in\{x,y,z\}} H_{n_i}\big(\sqrt{\alpha}\,(r_i - m_i)\big)\; e_{\boldsymbol{m}}(\boldsymbol{r}), \tag{33}$$

where H_n is the Hermite polynomial of degree n. This is the Hermite-Gaussian form used to evaluate molecular integrals (Helgaker et al., 2000). Second, a derivative in the center is a limit of differences of displaced copies. For a step $h > 0$, the forward difference is

$$\Delta_h^{\boldsymbol{n}} e_{\boldsymbol{m}} = \sum_{\boldsymbol{j}\leq\boldsymbol{n}} (-1)^{|\boldsymbol{n}|-|\boldsymbol{j}|} \binom{\boldsymbol{n}}{\boldsymbol{j}}\, e_{\boldsymbol{m}+h\boldsymbol{j}}, \qquad \binom{\boldsymbol{n}}{\boldsymbol{j}} = \prod_i \binom{n_i}{j_i}, \tag{34}$$

where $\boldsymbol{j} \le \boldsymbol{n}$ holds componentwise. It combines $(n_x+1)(n_y+1)(n_z+1)$ displaced copies of $e_{\boldsymbol{m}}$, all of which lie on the lattice

$$T_l = \{\boldsymbol{j} \in \mathbb{N}^3 : |\boldsymbol{j}| \le l\}, \qquad |T_l| = \tfrac{1}{6}(l+1)(l+2)(l+3),$$

whenever $|\boldsymbol{n}| \le l$.

Proposition B.2 (Gaussian-type orbitals are limits of splats)**.** *Let H^1 be the Sobolev space of square-integrable functions with square-integrable gradient, the natural space for the kinetic energy of Equation* (3)*.*

(i) For every $l \ge 0$, with $\boldsymbol{s} = \boldsymbol{r} - \boldsymbol{m}$,

$$\operatorname{span}\{\boldsymbol{s}^{\boldsymbol{n}} e^{-\alpha|\boldsymbol{s}|^2} : |\boldsymbol{n}| \le l\} = \operatorname{span}\{\partial_{\boldsymbol{m}}^{\boldsymbol{n}} e_{\boldsymbol{m}} : |\boldsymbol{n}| \le l\}.$$

(ii) For every multi-index $\boldsymbol{n}$, as $h \to 0$,

$$\left\| h^{-|\boldsymbol{n}|} \Delta_h^{\boldsymbol{n}} e_{\boldsymbol{m}} - \partial_{\boldsymbol{m}}^{\boldsymbol{n}} e_{\boldsymbol{m}} \right\|_{H^1} = \mathcal{O}(h).$$

(iii) Every function of a shell with angular momentum l and K primitives, for any contraction weights, is an H^1 limit of combinations of the $K\,|T_l|$ splats with the same exponents centered at $\boldsymbol{R}_a + h\boldsymbol{j}$, $\boldsymbol{j} \in T_l$. The same splats serve all $2l+1$ functions of the shell.

Proof. (i) In one dimension, the Rodrigues formula with $t = \sqrt{\alpha}(x-m)$ reads

$$H_n(t)\, e^{-t^2} = (-1)^n \frac{\mathrm{d}^n}{\mathrm{d}t^n} e^{-t^2},$$

and $\partial_m = -\sqrt{\alpha}\,\partial_t$ gives Equation (33). The three-dimensional case follows because $e_{\boldsymbol{m}}$ factorizes over coordinates. H_n has degree exactly n, so the Hermite polynomials and the monomials of degree at most n are related by an invertible triangular change of basis. Taking products over coordinates, the polynomials multiplying $e_{\boldsymbol{m}}$ on the two sides span the same space.

(ii) The map $\boldsymbol{m} \mapsto e_{\boldsymbol{m}}$ from $\mathbb{R}^3$ to H^1 is infinitely differentiable, because every center derivative is a polynomial times a Gaussian. For a smooth map into a Banach space, $h^{-|\boldsymbol{n}|}\Delta_h^{\boldsymbol{n}}$ equals $\partial_{\boldsymbol{m}}^{\boldsymbol{n}}$ plus a Taylor remainder bounded by h times the supremum of the derivatives of order $|\boldsymbol{n}|+1$ over the box spanned by the lattice points, which is finite.

(iii) A function of the shell is a solid harmonic of degree l times a contracted radial part, and a solid harmonic is a combination of monomials of degree l. By (i), each primitive is a finite combination of center derivatives of order at most l. By (ii), each of these is the H^1 limit of combinations of copies centered at $\boldsymbol{R}_a + h\boldsymbol{j}$, $\boldsymbol{j} \in T_l$. The contraction is a fixed finite sum over the K exponents, and the lattice does not depend on m. □

The inclusion extends from functions to energies. Suppose E is continuous in H^1 on orthonormal orbitals, as it is for Hartree–Fock and the local density approximation. Approximating each orbital within δ in H^1 moves the Löwdin-orthonormalized orbitals by $\mathcal{O}(\delta)$, because the retraction of Equation (11) is continuous where the Gram matrix is invertible. Hence, for a basis with shells (l_s, K_s),

$$\inf_{\Theta,\boldsymbol{C}} E \;\le\; E_{\mathrm{GTO}} \qquad \text{over clouds of size} \qquad M = \sum_s K_s\,|T_{l_s}|. \tag{35}$$

Proposition B.3 (The cost of the limit)**.** *Let $\boldsymbol{c}_h$ be the coefficient vector of $h^{-|\boldsymbol{n}|}\Delta_h^{\boldsymbol{n}} e_{\boldsymbol{m}}$ over the displaced copies, let $\boldsymbol{S}_h$ be their overlap matrix, and let*

$$\kappa_h = \frac{\sum_{\boldsymbol{j}} |(\boldsymbol{c}_h)_{\boldsymbol{j}}|\, \|e_{\boldsymbol{m}+h\boldsymbol{j}}\|_{L^2}}{\|h^{-|\boldsymbol{n}|}\Delta_h^{\boldsymbol{n}} e_{\boldsymbol{m}}\|_{L^2}}$$

be the relative condition number of the sum. Then, as $h \to 0$,

$$\begin{aligned} \|\boldsymbol{c}_h\| &\ge h^{-|\boldsymbol{n}|}, \\ \lambda_{\min}(\boldsymbol{S}_h) &= \mathcal{O}\big(h^{2|\boldsymbol{n}|}\big), \\ \kappa_h &= \Theta\big(h^{-|\boldsymbol{n}|}\big). \end{aligned}$$

Proof. By Equation (34), $\boldsymbol{c}_h = h^{-|\boldsymbol{n}|}\boldsymbol{v}$, where the entries of $\boldsymbol{v}$ are signed binomial coefficients and the entry for $\boldsymbol{j} = \boldsymbol{n}$ is one, so $\|\boldsymbol{v}\| \geq 1$. By Proposition B.2(ii) in L^2,

$$\begin{aligned}\boldsymbol{v}^\top \boldsymbol{S}_h \boldsymbol{v} &= \|\Delta_h^{\boldsymbol{n}} e_{\boldsymbol{m}}\|_{L^2}^2 \\ &= h^{2|\boldsymbol{n}|}\, \|\partial_{\boldsymbol{m}}^{\boldsymbol{n}} e_{\boldsymbol{m}}\|_{L^2}^2 + \mathcal{O}\big(h^{2|\boldsymbol{n}|+1}\big).\end{aligned}$$

The Rayleigh quotient gives $\lambda_{\min}(\boldsymbol{S}_h) \leq \boldsymbol{v}^\top \boldsymbol{S}_h \boldsymbol{v}/\|\boldsymbol{v}\|^2$, which is the second statement. For the third, all copies have the same norm $\|e_{\boldsymbol{m}}\|_{L^2}$, so the numerator of κ_h is $h^{-|\boldsymbol{n}|}\,\|\boldsymbol{v}\|_1\,\|e_{\boldsymbol{m}}\|_{L^2}$, while the denominator tends to $\|\partial_{\boldsymbol{m}}^{\boldsymbol{n}} e_{\boldsymbol{m}}\|_{L^2} > 0$. □

Proposition B.2 licenses two statements. First, the chart of Equation (8) needs no angular factor, since positive splats with signed coefficients reach every angular momentum. Second, Equation (35) shows that a cloud is at least as expressive as the Gaussian basis it replaces. The construction proves inclusion and is not the representation that the optimizer finds. It uses $|T_1| = 4$ splats per p primitive and $|T_2| = 10$ per d primitive, more than the $2l + 1$ functions of a shell, while the clouds of Section 5 reach lower energies than the Gaussian basis at matched function count. That gain comes from optimizing where the splats go.

Proposition B.3 shows that the limit is not attained by any finite cloud. Reproducing an angular factor exactly requires coincident splats with coefficients of size h^{-l}, and their overlap matrix becomes singular as h^{2l}. The occupied Gram matrix $\boldsymbol{G} = \boldsymbol{C}^\top \boldsymbol{S}\boldsymbol{C}$ of Section A.4 can stay well conditioned in this limit, because the coefficients compensate for the near-singular overlap. The compensation is a cancellation, however, and in floating-point arithmetic with unit roundoff u it carries a relative error of order $u\,\kappa_h$. A finite cloud therefore imitates angular momentum l only down to displacements h with $u\,h^{-l} \ll 1$, and this, not the floor of Equation (12), is the limit.

B.3 Floating-orbital models

Optimizing the positions and shapes of Gaussian basis functions is an old idea, and the question is why it did not become standard practice. This subsection shows that the earlier floating-orbital models are special cases of GS-DFT, and it states the four obstacles that stopped them as precise costs. Each obstacle is removed by a specific component of GS-DFT.

Gaussians entered quantum chemistry because their integrals have closed forms. Boys (1950) observed that the product of two Gaussians on different centers is a single Gaussian on a third, Equation (22), so every one- and two-electron integral reduces to elementary functions and the Boys function. The price is the shape. A Gaussian has no cusp at the nucleus and decays faster than an atomic orbital, so many Gaussians are needed per orbital, and standard basis sets pin fixed contractions of them to the atoms (Definition B.1).

Frost (1967) took the opposite route with the floating spherical Gaussian orbital (FSGO) model. It assigns one normalized spherical Gaussian to each doubly occupied orbital and minimizes the Hartree–Fock energy of the resulting determinant over the centers and widths, with no atom-centered basis. Vescelius & Frost (1974) and Cohen & Basch (1976) made the Gaussians ellipsoidal. Pederson et al. (1988) optimized floating Gaussians in DFT by simulated annealing and used the absence of Pulay terms to obtain Hellmann–Feynman forces without corrections, the result of Proposition A.3. Explicitly correlated Gaussians pushed nonlinear optimization of Gaussian parameters to spectroscopic accuracy, at a cost that confines them to a handful of particles (Mitroy et al., 2013). More recently, Tamayo-Mendoza et al. (2018) used automatic differentiation to optimize basis exponents and centers in Hartree–Fock on small molecules.

Proposition B.4 (FSGO is GS-DFT at $M = N_{\mathrm{occ}}$)**.** *Let $M = N_{\mathrm{occ}}$ and let $\boldsymbol{C}$ be invertible. Then the density matrix of GS-DFT does not depend on $\boldsymbol{C}$,*

$$\boldsymbol{P} = 2\,\boldsymbol{S}(\Theta)^{-1},$$

and the GS-DFT energy is a function of Θ alone. With isotropic splats and E_{xc} set to exact exchange, it is the FSGO energy of Frost (1967). With anisotropic splats, it is the ellipsoidal model of Vescelius & Frost (1974).

Proof. With $\boldsymbol{G} = \boldsymbol{C}^\top \boldsymbol{S}\boldsymbol{C}$ and the Löwdin coefficients of Equation (11),

$$\begin{aligned}\boldsymbol{P} &= 2\,\bar{\boldsymbol{C}}\bar{\boldsymbol{C}}^\top \\ &= 2\,\boldsymbol{C}\,(\boldsymbol{C}^\top \boldsymbol{S}\boldsymbol{C})^{-1}\boldsymbol{C}^\top \\ &= 2\,\boldsymbol{C}\,\boldsymbol{C}^{-1}\boldsymbol{S}^{-1}\boldsymbol{C}^{-\top}\boldsymbol{C}^\top \\ &= 2\,\boldsymbol{S}^{-1}.\end{aligned}$$

The density $\rho = \sum_{\mu\nu} \boldsymbol{P}_{\mu\nu}\, g_\mu g_\nu$, and hence every term of the energy, then depends on Θ only. It is the density of the determinant built from the N_{occ} Gaussians themselves, whose energy is invariant under any invertible recombination of its orbitals. That is the determinant FSGO optimizes, with spherical Gaussians in the original model and ellipsoidal ones in its extension. □

Proposition B.4 places the whole lineage in one corner of the GS-DFT family: the minimal cloud, $M = N_{\mathrm{occ}}$, with the coefficients reduced to a gauge. GS-DFT keeps the model and moves along the other axis, $M > N_{\mathrm{occ}}$. Four obstacles made that axis unreachable. Write $D = M(9 + N_{\mathrm{occ}})$ for the number of free parameters and T for the number of optimization steps.

(i) *Integral cost.* A fixed basis computes its electron-repulsion tensor once and reuses it at every iteration. In a floating basis every entry depends on Θ, so the two-electron term costs

$$\mathcal{O}(M^4) \text{ integrals per step,} \qquad \mathcal{O}(TM^4) \text{ in total.}$$

A fixed auxiliary basis $\{\varphi_a\}$ pinned to the atoms does not remove this cost without an accuracy penalty, because by Equation (21) its error is $\frac{1}{2}\|\rho - \Pi_\varphi \rho\|_{\mathrm{C}}^2$, and nothing controls it once the cloud, and with it ρ, moves away from $\mathrm{span}\{\varphi_a\}$.

(ii) *Gradients.* Optimizing Θ needs the gradient of the energy with respect to all D parameters. Finite differences cost $D + 1$ energy evaluations per gradient, and analytic gradients without automatic differentiation require a hand-derived derivative for every integral class.

(iii) *Optimization at scale.* The energy is nonconvex in Θ. Newton-type methods need the $D \times D$ Hessian,

$$\mathcal{O}(D^2) \text{ memory and } \mathcal{O}(D^3) \text{ time per step,}$$

which is infeasible for the clouds of Section 5.

(iv) *Linear dependence.* Freely moving functions approach each other, $\lambda_{\min}(\boldsymbol{S}) \to 0$ (Proposition B.3), and the eigendecomposition gradient of the orthonormalization contains factors $(\lambda_i - \lambda_j)^{-1}$ that diverge at degeneracies.

Table 5 pairs each obstacle with the component of GS-DFT that removes it. The adaptive density fit of Section 4 rebuilds the auxiliary basis from the cloud, so its error is bounded by Proposition A.5 rather than growing with the motion of the cloud, and its per-step cost is quadratic (Algorithm 1). Reverse-mode automatic differentiation evaluates the full gradient at a cost bounded by a constant multiple of one energy evaluation, independent of D (Baydin et al., 2018). Adam (Kingma & Ba, 2015) needs $\mathcal{O}(D)$ memory and time per step, the regime in which 3DGS optimizes millions of Gaussians (Kerbl et al., 2023). The regularized eigensolve bounds the backward pass at degeneracies (Proposition A.6).

Table 5: Obstacles that confined floating-Gaussian models to $M = N_{\mathrm{occ}}$, and the component of GS-DFT that removes each.

Obstacle	Earlier cost	Component of GS-DFT	Cost or bound
(i) Integrals	$\mathcal{O}(M^4)$ per step	Adaptive density fitting	Algorithm 1, Proposition A.5
(ii) Gradients	$D + 1$ evaluations	Reverse-mode differentiation	$\mathcal{O}(1)$ evaluations
(iii) Scale	$\mathcal{O}(D^3)$ per step	Adam	$\mathcal{O}(D)$ per step
(iv) Dependence	unbounded gradient	Regularized eigensolve	Proposition A.6

None of the obstacles is a limit of the representation, which Proposition B.2 shows is at least as expressive as a fixed basis. They are limits of the numerics, and they are what the related-work section means by inaccurate and unscalable: the earlier models were inaccurate because they were minimal, and they stayed minimal because (i) to (iv) made $M > N_{\mathrm{occ}}$ too expensive.

B.4 Correspondence with 3D Gaussian Splatting

Section 2 writes 3DGS and DFT as two instances of one optimization problem over Gaussian primitives, Equations (4) and (7). The correspondence is what lets GS-DFT inherit the 3DGS toolkit, and it is also easy to overstate. This subsection proves what carries over exactly and states precisely where the two problems differ.

The chart carries over exactly. Kerbl et al. (2023) parameterize the covariance of each Gaussian by a unit quaternion $\boldsymbol{q}$ and a log-scale vector $\boldsymbol{\sigma} \in \mathbb{R}^3$,

$$\boldsymbol{\Sigma} = \boldsymbol{U}(\boldsymbol{q})\,\mathrm{diag}(e^{2\boldsymbol{\sigma}})\,\boldsymbol{U}(\boldsymbol{q})^\top.$$

Since $\boldsymbol{A} = \boldsymbol{\Sigma}^{-1}$ and $\boldsymbol{U}(\boldsymbol{q})$ is orthogonal,

$$\begin{aligned}\boldsymbol{A} &= \boldsymbol{U}(\boldsymbol{q})\,\mathrm{diag}(e^{-2\boldsymbol{\sigma}})\,\boldsymbol{U}(\boldsymbol{q})^\top,\\ \boldsymbol{l} &= -2\boldsymbol{\sigma},\end{aligned}$$

so the 3DGS chart and the chart of Equation (8) are the same map up to a fixed linear change of the log-scales. Both are unconstrained, every point is a valid Gaussian, and both are equivariant (Proposition A.1). We parameterize the precision because it is what enters the closed-form integrals, through the product precision $\boldsymbol{A}_\mu + \boldsymbol{A}_\nu$ of Equations (9) and (22). Both methods optimize all primitives jointly with Adam, although 3DGS assigns each parameter group its own learning rate while GS-DFT uses a single cosine-scheduled rate (Section C.5). Table 6 pairs the remaining components.

Table 6: Components of 3DGS and their counterparts in GS-DFT.

	3DGS	GS-DFT
Primitive	anisotropic Gaussian	anisotropic Gaussian
Shape chart	covariance, quaternion and log-scales	precision, quaternion and log-eigenvalues
Weights	opacity and color, nonnegative	orbital coefficients $\boldsymbol{C}$, signed
Channels	color channels	occupied orbitals
Represented signal	radiance along rays	orbitals ψ_k and density ρ
Forward map	projection and alpha compositing	closed-form integrals and quadrature
Objective	photometric loss against views	variational energy, no data
Constraint	none	orthonormality, via Equation (11)
Initialization	structure-from-motion points	chemically informed, from nuclei
Cloud size	adaptive, by densification and pruning	fixed M
Optimizer	Adam	Adam

The model is linear where 3DGS is not. 3DGS renders a pixel $\boldsymbol{x}$ by sorting the Gaussians along its ray and alpha compositing their colors $\boldsymbol{c}_i$,

$$\begin{aligned}\mathcal{R}(\boldsymbol{x}) &= \sum_i \boldsymbol{c}_i\, a_i(\boldsymbol{x}) \prod_{j<i} \big(1 - a_j(\boldsymbol{x})\big),\\ a_i(\boldsymbol{x}) &= o_i\, G_i^{\mathrm{2D}}(\boldsymbol{x}),\end{aligned} \tag{36}$$

with opacity $o_i \in (0,1)$ and projected footprint $G_i^{\mathrm{2D}} \le 1$. Expanding the product,

$$\mathcal{R}(\boldsymbol{x}) = \sum_i \boldsymbol{c}_i\, a_i(\boldsymbol{x}) + \mathcal{O}\big(\max_j a_j(\boldsymbol{x})^2\big),$$

so the linear superposition of Equation (7) is the first-order, low-opacity limit of Equation (36). In GS-DFT the orbitals are exactly linear in $\boldsymbol{C}$ at fixed Θ and the density is exactly quadratic.

The weights must be signed. 3DGS weights are nonnegative because radiance is. Orbitals cannot be built that way.

Proposition B.5 (Signed coefficients are necessary)**.** *If $N_{\mathrm{occ}} \ge 2$, no two orthonormal orbitals can both be nonnegative combinations of splats.*

Proof. Every splat is strictly positive everywhere, so a nonnegative, nonzero combination of splats is strictly positive everywhere. If ψ_1 and ψ_2 were both of this form, then $\psi_1\psi_2 > 0$ everywhere and

$$\int \mathrm{d}^3\boldsymbol{r}\ \psi_1(\boldsymbol{r})\,\psi_2(\boldsymbol{r}) > 0,$$

which contradicts orthogonality. □

At most one orbital can be sign-definite, and every other orbital needs coefficients of both signs. This is the nodal structure that Proposition B.2 builds from displaced splats of opposite sign.

The objective couples every pair of primitives. In 3DGS each pixel depends only on the Gaussians whose footprints cover it, and the loss is a sum over pixels. In GS-DFT, the Hartree energy couples every pair of splat products through the Coulomb kernel,

$$E_H = \tfrac{1}{2}\sum_{\mu\nu\lambda\sigma} \boldsymbol{P}_{\mu\nu}\boldsymbol{P}_{\lambda\sigma}\,(g_\mu g_\nu \,|\, g_\lambda g_\sigma),$$

and this coupling does not decay to zero with distance. By the Gaussian product theorem, $g_\mu g_\nu$ carries charge $\boldsymbol{S}_{\mu\nu}$ at the center $\boldsymbol{m}_{\mu\nu}$ of Equation (22), and as the two products separate,

$$(g_\mu g_\nu \,|\, g_\lambda g_\sigma) = \frac{\boldsymbol{S}_{\mu\nu}\,\boldsymbol{S}_{\lambda\sigma}}{|\boldsymbol{m}_{\mu\nu} - \boldsymbol{m}_{\lambda\sigma}|}\big(1 + o(1)\big).$$

The interaction falls off only as the inverse distance, so it cannot be truncated by locality the way a rendering footprint can. This is why GS-DFT needs the density fit of Section A.3. The orthonormality constraint couples every pair of orbitals in the same global way, which is why it needs the retraction of Section A.4.

Nothing is projected. Splatting in graphics is the projection of each Gaussian onto the image plane. The elliptical weighted average filter of Zwicker et al. (2001) replaces the camera map by its Jacobian $\boldsymbol{J}_{\mathrm{cam}}$ at the Gaussian's center, so that the projection stays Gaussian with covariance

$$\boldsymbol{\Sigma}' = \boldsymbol{J}_{\mathrm{cam}}\,\boldsymbol{W}\,\boldsymbol{\Sigma}\,\boldsymbol{W}^\top\boldsymbol{J}_{\mathrm{cam}}^\top,$$

for a viewing transformation $\boldsymbol{W}$. The projection is therefore exact only to first order in the camera map. GS-DFT integrates in three dimensions, the overlap, kinetic and Coulomb integrals are exact closed forms, and the only approximation in the forward map is the quadrature of E_{xc}. The name refers to the primitive and the chart, not to a rendering step.

There is no data. 3DGS solves an inverse problem. It fits a set of observed views $\{I_v\}$, and its quality is measured on views held out from the fit, so the training loss can decrease while the held-out error increases. GS-DFT solves a forward problem. For exact integrals and orthonormal orbitals, the Rayleigh–Ritz principle bounds every parameter value by the complete-basis energy E_{CBS} of the same functional,

$$E(\Theta, \boldsymbol{C}) \ \geq\ E_{\mathrm{CBS}} \qquad \text{for all } (\Theta, \boldsymbol{C}),$$

so a lower energy is always a better solution, and there is no held-out set and no generalization gap. With density fitting, the bound holds up to the gap of Equation (21). The price is that there is no data to initialize from, which is why GS-DFT places its initial cloud from the nuclei instead of from reconstructed points.

C Implementation

This appendix describes GS-DFT as it runs. Section C.1 gives the software stack and the numerical conventions, and Section C.2 derives every integral of the energy. Section C.3 gives the cost and memory of each operation, and Section C.4 explains how one system is spread over several devices. Section C.5 lists the training settings, Section C.6 the evaluation protocol behind every reported number, and Section C.7 the hardware, software and code layout.

C.1 SOFTWARE AND NUMERICS

GS-DFT uses `dftax` (Guzmán-Cordero, 2026) as its base, because it implements Kohn–Sham DFT natively in JAX (Bradbury et al., 2018; Kidger & Garcia, 2021). The engine supplies the exchange–correlation functionals, the Becke quadrature grids, the Boys function, and the fixed-basis Gaussian reference calculations. Everything specific to splats lives in the `gs-dft` package:[1] the chart of Equation (8), the splat integrals, the adaptive density fit, screening, the regularized eigensolve, and multi-device sharding. PySCF is used only as an external reference, for the calculations listed in Section C.6.

All arithmetic is in double precision, which the package enables globally when it is imported. The occupations are frozen, and every term of Equation (3) is a differentiable function of the parameters $(\Theta, \boldsymbol{C})$. One reverse-mode pass therefore returns the gradient with respect to all of them, including the basis gradients that flow through the Löwdin retraction of Equation (11). Three operations carry hand-written reverse passes: the regularized eigensolve of Section A.4, the streamed Coulomb contractions, and the Coulomb quadrature of Section C.2. Each replaces an automatically generated backward pass that would either store one intermediate per quadrature node or divide by eigenvalue gaps.

The chart is implemented as written in Equation (8), so the precision and its derived quantities are closed forms in $(\boldsymbol{q}, \boldsymbol{l})$,

$$\begin{aligned} \boldsymbol{A} &= \boldsymbol{U}(\boldsymbol{q})\,\mathrm{diag}(e^{\boldsymbol{l}})\,\boldsymbol{U}(\boldsymbol{q})^\top, \\ \det \boldsymbol{A} &= e^{\sum_i \ell_i}, \\ g_\theta(\boldsymbol{r}) &= \det(\boldsymbol{A}/\pi)^{1/4}\,\exp\!\big\{-\tfrac{1}{2}(\boldsymbol{r}-\boldsymbol{m})^\top \boldsymbol{A}\,(\boldsymbol{r}-\boldsymbol{m})\big\}, \end{aligned}$$

and no matrix decomposition of a splat's own precision is ever needed. Every 3×3 determinant, inverse and linear solve elsewhere in the code uses the adjugate formula, which is exact, branch-free and faster than a batched library call at this size.

C.2 INTEGRALS

Every term of Equation (3) except the exchange–correlation energy reduces to a Gaussian integral over a product of two splats, followed, for the Coulomb terms, by a one-dimensional quadrature. The exchange–correlation energy is a quadrature on a grid. The building block is the product of Equation (22),

$$g_\mu(\boldsymbol{r})\,g_\nu(\boldsymbol{r}) = N_\mu N_\nu\,\boldsymbol{K}_{\mu\nu}\,\exp\!\big\{-\tfrac{1}{2}(\boldsymbol{r}-\boldsymbol{m}_{\mu\nu})^\top \boldsymbol{A}_{\mu\nu}\,(\boldsymbol{r}-\boldsymbol{m}_{\mu\nu})\big\}, \tag{37}$$

with $N_\mu = \det(\boldsymbol{A}_\mu/\pi)^{1/4}$, the prefactor $\boldsymbol{K}_{\mu\nu}$ of Equation (10), and

$$\begin{aligned} \boldsymbol{A}_{\mu\nu} &= \boldsymbol{A}_\mu + \boldsymbol{A}_\nu, \\ \boldsymbol{m}_{\mu\nu} &= \boldsymbol{A}_{\mu\nu}^{-1}(\boldsymbol{A}_\mu \boldsymbol{m}_\mu + \boldsymbol{A}_\nu \boldsymbol{m}_\nu). \end{aligned}$$

Overlap and kinetic energy. Integrating Equation (37) gives the overlap of Equation (9),

$$\begin{aligned} \boldsymbol{S}_{\mu\nu} &= N_\mu N_\nu\,\boldsymbol{K}_{\mu\nu}\,(2\pi)^{3/2}\det(\boldsymbol{A}_{\mu\nu})^{-1/2} \\ &= 2^{3/2}\,\frac{(\det \boldsymbol{A}_\mu \det \boldsymbol{A}_\nu)^{1/4}}{\det(\boldsymbol{A}_\mu + \boldsymbol{A}_\nu)^{1/2}}\,\boldsymbol{K}_{\mu\nu} = \sqrt{\frac{\det\sqrt{\boldsymbol{A}_\mu \boldsymbol{A}_\nu}}{\det\big(\tfrac{1}{2}(\boldsymbol{A}_\mu + \boldsymbol{A}_\nu)\big)}}\,\boldsymbol{K}_{\mu\nu}\,, \end{aligned}$$

where the last line suggests an algebraic interpretation of the prefactor – it is a determinant of the ratio of the geometric and algebraic means of the two precision matrices. For the kinetic energy, $\boldsymbol{\nabla} g_\mu(\boldsymbol{r}) = -\boldsymbol{A}_\mu(\boldsymbol{r}-\boldsymbol{m}_\mu)\,g_\mu(\boldsymbol{r})$. Writing $\boldsymbol{r} = \boldsymbol{m}_{\mu\nu} + \boldsymbol{x}$, with $\boldsymbol{a} = \boldsymbol{m}_{\mu\nu} - \boldsymbol{m}_\mu$ and $\boldsymbol{b} = \boldsymbol{m}_{\mu\nu} - \boldsymbol{m}_\nu$, the integrand is a quadratic form in $\boldsymbol{x}$ under a Gaussian of covariance $\boldsymbol{A}_{\mu\nu}^{-1}$, and the cross terms vanish by symmetry. Hence

$$\begin{aligned} \boldsymbol{T}_{\mu\nu} &= \tfrac{1}{2}\int \mathrm{d}^3\boldsymbol{r}\;\boldsymbol{\nabla} g_\mu \cdot \boldsymbol{\nabla} g_\nu \\ &= \tfrac{1}{2}\,\boldsymbol{S}_{\mu\nu}\Big[\,\mathrm{tr}\big(\boldsymbol{A}_\mu \boldsymbol{A}_\nu \boldsymbol{A}_{\mu\nu}^{-1}\big) + \boldsymbol{a}^\top \boldsymbol{A}_\mu \boldsymbol{A}_\nu\,\boldsymbol{b}\Big]. \end{aligned}$$

[1] Anonymized repository: `https://anonymous.4open.science/r/gs-dft-26FC`.

The Coulomb kernel. The Coulomb kernel has no Gaussian closed form, but it is a superposition of Gaussians,

$$\frac{1}{|\boldsymbol{r}|} = \frac{2}{\sqrt{\pi}} \int_0^\infty e^{-t^2|\boldsymbol{r}|^2}\,\mathrm{d}t. \tag{38}$$

Let p and q be two Gaussian charge distributions with charges Q_p and Q_q, centers $\boldsymbol{m}_p$ and $\boldsymbol{m}_q$, and precisions $\boldsymbol{A}_p$ and $\boldsymbol{A}_q$. Substituting Equation (38) into the Coulomb inner product of Equation (19) gives

$$(p\,|\,q) = Q_p Q_q\,\frac{2}{\sqrt{\pi}} \int_0^\infty \det\!\big(\mathbb{1} + 2t^2\boldsymbol{\Sigma}_{pq}\big)^{-1/2} \exp\!\Big\{ -t^2\,\boldsymbol{\Delta}^\top \big(\mathbb{1} + 2t^2\boldsymbol{\Sigma}_{pq}\big)^{-1}\boldsymbol{\Delta}\Big\}\,\mathrm{d}t, \tag{39}$$

with

$$\boldsymbol{\Sigma}_{pq} = \boldsymbol{A}_p^{-1} + \boldsymbol{A}_q^{-1},$$
$$\boldsymbol{\Delta} = \boldsymbol{m}_p - \boldsymbol{m}_q.$$

The derivation uses one fact: if $\boldsymbol{x}$ and $\boldsymbol{y}$ are distributed as the normalized densities p/Q_p and q/Q_q, then $\boldsymbol{x} - \boldsymbol{y} - \boldsymbol{\Delta}$ is a centered Gaussian with covariance $\boldsymbol{\Sigma}_{pq}$. The inner integral is then the expectation of $e^{-t^2|\boldsymbol{z}+\boldsymbol{\Delta}|^2}$ over $\boldsymbol{z} \sim \mathcal{N}(0, \boldsymbol{\Sigma}_{pq})$, which is the integrand of Equation (39). For a splat product, the charge is the overlap, $Q = \boldsymbol{S}_{\mu\nu}$, which is the charge used in the far-field argument of Section B.4.

Every Coulomb quantity of the method is an instance of Equation (39): the auxiliary metric $\boldsymbol{Q}$ and the fit vector $\boldsymbol{t}$ of Equation (14), whose auxiliary functions are splat products, and the exact Hartree energy used for evaluation. In general $\boldsymbol{\Sigma}_{pq}$ is a 3×3 matrix, and its determinant and inverse are adjugate closed forms. Two limits are elementary. If $\boldsymbol{\Sigma}_{pq} = s\,\mathbb{1}$ is isotropic, the integral is

$$(p\,|\,q) = Q_p Q_q\,\frac{\operatorname{erf}\!\big(|\boldsymbol{\Delta}|/\sqrt{2s}\big)}{|\boldsymbol{\Delta}|},$$

which is the Boys function of order zero. A nucleus is the limit $\boldsymbol{A}_q \to \infty$ of a Gaussian, with $\boldsymbol{\Sigma}_{pq} = \boldsymbol{A}_{\mu\nu}^{-1}$, $Q_q = -Z_a$ and $\boldsymbol{m}_q = \boldsymbol{R}_a$, which gives the nuclear attraction.

The half-line integral in Equation (39) is mapped to the unit interval by $t = s/(1-s)$ and evaluated with a 20-node Gauss–Legendre rule,

$$\begin{aligned}\int_0^\infty f(t)\,\mathrm{d}t &= \int_0^1 f\Big(\frac{s}{1-s}\Big)\,\frac{\mathrm{d}s}{(1-s)^2} \\ &\approx \sum_{n=1}^{20} \frac{w_n}{(1-s_n)^2}\,f\Big(\frac{s_n}{1-s_n}\Big),\end{aligned}$$

with nodes s_n and weights w_n of the rule on $[0,1]$. Range-separated hybrids such as ωB97M-V need the long-range kernel as well, and it is the same superposition over a finite interval,

$$\frac{\operatorname{erf}(\omega|\boldsymbol{r}|)}{|\boldsymbol{r}|} = \frac{2}{\sqrt{\pi}} \int_0^\omega e^{-t^2|\boldsymbol{r}|^2}\,\mathrm{d}t.$$

The integrand of Equation (39) is unchanged, and only the node table differs: a 12-node Gauss–Legendre rule on $[0,\omega]$ replaces the half-line rule. The node loop is unrolled into one fused kernel, and its reverse pass is written analytically, so that no per-node intermediate is stored.

Exchange–correlation. The exchange–correlation energy is the only term evaluated on a grid. The density, its gradient and the kinetic-energy density are evaluated exactly at the points $\boldsymbol{r}_g$ of a Becke grid with weights w_g,

$$\rho(\boldsymbol{r}_g) = \sum_{\mu\nu} \boldsymbol{P}_{\mu\nu}\,g_\mu(\boldsymbol{r}_g)\,g_\nu(\boldsymbol{r}_g),$$
$$E_{\mathrm{xc}} \approx \sum_g w_g\,e_{\mathrm{xc}}\big(\rho(\boldsymbol{r}_g),\,|\boldsymbol{\nabla}\rho(\boldsymbol{r}_g)|^2,\,\tau(\boldsymbol{r}_g)\big),$$

Table 7: Leading-order cost of one energy-and-gradient evaluation. The factorization of the auxiliary metric is paid once per refresh and appears amortized over the K steps it serves.

Operation	Algorithm	Compute	Memory
Overlap and kinetic, $\boldsymbol{SC}$ and $\boldsymbol{TC}$	Dense row blocks	$\mathcal{O}(M^2 N_{\mathrm{occ}})$	$\mathcal{O}(M N_{\mathrm{occ}})$
Occupied Gram and retraction, Equation (12)	Eigensolve of $\boldsymbol{G} = \boldsymbol{C}^\top \boldsymbol{SC}$	$\mathcal{O}(M N_{\mathrm{occ}}^2 + N_{\mathrm{occ}}^3)$	$\mathcal{O}(N_{\mathrm{occ}}^2)$
Nuclear attraction	Retained pairs	$\mathcal{O}(Mk N_{\mathrm{at}})$	$\mathcal{O}(Mk)$
Fitted Hartree, Equation (14)	Fit vector over retained pairs	$\mathcal{O}(Mk N_{\mathrm{aux}} + N_{\mathrm{aux}}^2 + N_{\mathrm{aux}}^3/K)$	$\mathcal{O}(N_{\mathrm{aux}}^2)$
Fitted exchange, hybrids only	One fit vector per orbital pair	$\mathcal{O}(Mk N_{\mathrm{aux}} N_{\mathrm{occ}}^2 + N_{\mathrm{aux}}^2 N_{\mathrm{occ}}^2 + N_{\mathrm{aux}}^3)$	$\mathcal{O}(N_{\mathrm{aux}}^2)$
Exchange–correlation	Chunked grid	$\mathcal{O}(N_g M N_{\mathrm{occ}})$	$\mathcal{O}(N_{g,\mathrm{c}} M)$
VV10 nonlocal correlation, ωB97M-V only	Chunked double grid sum	$\mathcal{O}(N_g^2)$	$\mathcal{O}(N_{g,\mathrm{c}} N_g)$
Exact Hartree, evaluation only	Streamed product pairs	$\mathcal{O}(M^4)$	$\mathcal{O}(M^2)$
Nuclear forces, once at the end	Reverse pass in $\boldsymbol{R}$	$\mathcal{O}(Mk N_{\mathrm{at}} + N_{\mathrm{at}}^2)$	$\mathcal{O}(Mk)$

where e_{xc} is the energy density of the functional supplied by `dftax`. Generalized-gradient functionals use ρ and $|\boldsymbol{\nabla}\rho|^2$, and meta-GGAs such as r^{2}SCAN also use the kinetic-energy density τ. The VV10 nonlocal correlation of ωB97M-V (Vydrov & Van Voorhis, 2010) is a double sum over the grid,

$$E_{\mathrm{c}}^{\mathrm{nl}} \approx \sum_g w_g \rho_g \Big[\beta + \tfrac{1}{2} \sum_{g'} w_{g'} \rho_{g'}\, \Phi_{gg'}\Big],$$
$$\Phi_{gg'} = -\frac{3}{2\, u_{gg'}\, u_{g'g}\, (u_{gg'} + u_{g'g})},$$
$$u_{gg'} = \omega_0(\boldsymbol{r}_g)\, |\boldsymbol{r}_g - \boldsymbol{r}_{g'}|^2 + \kappa(\boldsymbol{r}_g),$$

with $\rho_g = \rho(\boldsymbol{r}_g)$ and the local functions ω_0 and κ of VV10. It is streamed over chunks of the outer grid, and its reverse pass is written analytically, because automatic differentiation would store the $\mathcal{O}(N_g^2)$ pair tensor that the chunking avoids. The grid is processed in chunks, so the values of all splats at all grid points are never stored at once.

C.3 Complexity and memory

Table 7 lists the leading-order cost of every operation in one energy-and-gradient evaluation, in the following symbols:

M : splats,	N_{occ} : occupied orbitals,	N_{at} : nuclei,
N_{aux} : auxiliary functions,	k : retained neighbors per splat,	N_g : grid points,
K : refresh period,	$N_{g,\mathrm{c}}$: grid points per chunk.	

The number of retained pairs is $|\mathcal{P}| = \mathcal{O}(Mk)$, and every Coulomb entry carries the constant factor of the 20 quadrature nodes of Section C.2. Reverse-mode differentiation multiplies each cost by a constant and leaves every order unchanged.

The training step costs

$$\mathcal{O}\Big(M^2 N_{\mathrm{occ}} + Mk N_{\mathrm{aux}} + N_{\mathrm{aux}}^2 + N_g M N_{\mathrm{occ}} + \frac{N_{\mathrm{aux}}^3}{K}\Big) \tag{40}$$

for semilocal functionals. The last term is the one cubic operation of the method, the factorization of the auxiliary metric, amortized over K steps. Hybrid functionals add the exchange row, whose metric solve is repeated at every step, which is why they are run only on the smaller systems of Section 5.

Three rows deserve comment. The overlap and kinetic products are dense, not screened. Discarding pairs from $\boldsymbol{S}$ and $\boldsymbol{T}$ makes the occupied Gram matrix and the kinetic energy indefinite, and a retraction built on an indefinite Gram matrix is not an orthonormalization. They are still never stored, because both are needed only through $\boldsymbol{SC}$ and $\boldsymbol{TC}$, which are contracted row block by row block. The nuclear attraction, by contrast, enters only as a trace against $\boldsymbol{P}$, has no positivity requirement, and is screened. Finally, the exact Hartree energy is never used in training. It serves the final evaluations and the exact-Coulomb columns of the ablations.

The memory column is what sets the size limit of Section 5, and it is controlled by one rule: no operator is materialized. The fit vector is accumulated over chunks of the retained pair list, the grid is processed in chunks, and each streamed block is recomputed in the backward pass rather than stored.

The gradient therefore holds one block at a time, exactly as the energy does. Peak memory is then set by two terms: the Cholesky factor of the auxiliary metric, $\mathcal{O}(N_{\mathrm{aux}}^2)$, and the temporaries of one chunk of the fit-vector stream, $\mathcal{O}(N_{\mathrm{aux}}\, c)$ for a chunk of c retained pairs. At the protein scale the second term dominates, so the chunk size is the memory knob: on insulin, reducing c from 2048 to 512 halves the peak memory of a step, from 35.4 to 17.6 GB, costs 2.8% more time, and leaves the energy unchanged to twelve digits, because chunking reorders the same arithmetic. The protein-scale runs use $c = 512$. The neighbor degree k saturates as the cloud grows, because each splat overlaps only splats within a fixed distance, so the screened rows grow linearly in M at fixed k.

C.4 Multi-device sharding

One system is spread over n devices by sharding the work and replicating the parameters. The parameters $(\Theta, \boldsymbol{C})$ occupy $\mathcal{O}(MN_{\mathrm{occ}})$ memory, a fraction of the per-device budget at every size in Section 5, so partitioning them would buy little and would force communication inside every contraction. Instead, each expensive term of Table 7 is written as a sum over an index set, and the index set is partitioned.

What is partitioned. Three index sets are split into n blocks, one per device d:

$$\begin{aligned} \{1,\dots,M\} &= R_1 \cup \cdots \cup R_n && \text{rows of } \boldsymbol{SC} \text{ and } \boldsymbol{TC}, \\ \mathcal{P} &= \mathcal{P}_1 \cup \cdots \cup \mathcal{P}_n && \text{retained pairs}, \\ \{\boldsymbol{r}_g\} &= \mathcal{G}_1 \cup \cdots \cup \mathcal{G}_n && \text{grid points}. \end{aligned}$$

Each term is then a sum of per-device partials followed by one all-reduce. For the occupied Gram matrix, the fit vector and the exchange–correlation energy,

$$\begin{aligned} \boldsymbol{G} &= \sum_{d=1}^{n} \boldsymbol{C}_{R_d}^{\top} (\boldsymbol{SC})_{R_d}, \\ \boldsymbol{t} &= \sum_{d=1}^{n} \boldsymbol{t}^{(d)}, \\ \boldsymbol{t}_a^{(d)} &= (\rho_{\mathcal{P}_d} \,|\, h_a), \\ E_{\mathrm{xc}} &\approx \sum_{d=1}^{n} \sum_{g \in \mathcal{G}_d} w_g\, e_{\mathrm{xc}}(\boldsymbol{r}_g), \end{aligned}$$

and the kinetic and nuclear-attraction energies are split in the same way over rows and pairs. The small operations that follow the all-reduce run replicated on every device: the retraction of Equation (12) on the $N_{\mathrm{occ}} \times N_{\mathrm{occ}}$ matrix $\boldsymbol{G}$, and the metric solve of Equation (14) on the length-N_{aux} vector $\boldsymbol{t}$.

Gradients. The adjoint of an all-reduce is an all-reduce, so reverse-mode differentiation through the sharded energy produces the sharded gradient with no extra code. The sharded value and gradient therefore equal the single-device ones up to the order of floating-point summation.

Load balance. The pair list is padded to a fixed length, twice the number of pairs retained at initialization, so that its shape stays constant under compilation. The padding must not overflow: pairs that do not fit are dropped, and the energy of the truncated sum is not a converged result, which a padding of 1.25 produces on alanine dipeptide. A contiguous split of the padded list would hand the last devices only padding. The list is instead dealt round-robin: with $|\mathcal{P}| = n\ell$ padded entries, entry $jn + d$ of the original order becomes entry $d\ell + j$ of device d's block. Each device then receives the same number of valid pairs up to one, stored at the front of its block. Every consumer of the list is an order-independent sum, so the reordering changes the work distribution and not the energy.

Replica synchronization. After every optimizer step, the replicated parameters are averaged across devices,

$$(\Theta, \boldsymbol{C}) \;\leftarrow\; \frac{1}{n} \sum_{d=1}^{n} (\Theta, \boldsymbol{C})^{(d)}.$$

In exact arithmetic the copies are identical and the average is a no-op. In floating point they are not, because the replicated operations are recomputed on each device and can round differently. The forward retraction is insensitive to such differences, since $f(\boldsymbol{G})$ is a smooth matrix function (Equation (25)). Its regularized backward pass is not, because it switches between the divided difference and the midpoint rule at $|\lambda_i - \lambda_j| = \delta\lambda_{\max}$ and drops floored modes (Proposition A.6). Both switches are discontinuous in the eigenvalues, so Gram matrices that agree to the last bit can receive different gradients, and without the average the copies separate over the course of training. The average costs one all-reduce of $\mathcal{O}(MN_{\mathrm{occ}})$ numbers per step.

Cost per device. The sharded terms of Equation (40) divide by n, and the replicated ones do not,

$$\mathcal{O}\Big(\frac{M^2N_{\mathrm{occ}} + MkN_{\mathrm{aux}} + N_gMN_{\mathrm{occ}}}{n} + N_{\mathrm{occ}}^3 + N_{\mathrm{aux}}^2 + \frac{N_{\mathrm{aux}}^3}{K}\Big).$$

Communication per step is the all-reduce of $\boldsymbol{G}$, $\boldsymbol{t}$, a few scalars, and the parameter average. The Cholesky factor of the auxiliary metric is replicated, and the chunk of the fit-vector stream is a per-device constant, so both memory terms of Section C.3 are the same on every device. Adding devices shortens each device's share of the work but does not reduce its peak memory. The same code runs across nodes through the multi-process runtime of JAX. The sharded path does not yet evaluate the kinetic-energy density, so meta-GGAs run on one device.

C.5 Training recipe

Optimizer. All runs use Adam with global-norm gradient clipping, a linear warmup and a cosine decay,

$$\eta_t = \eta_{\max}\Big(0.01 + 0.99\,\frac{t}{T_w}\Big) \qquad t < T_w,$$
$$\eta_t = \eta_{\max}\Big(0.01 + 0.99\cdot\tfrac{1}{2}\Big[1 + \cos\Big(\pi\,\frac{t - T_w}{T - T_w}\Big)\Big]\Big) \qquad t \geq T_w,$$

with peak rate $\eta_{\max} = 10^{-2}$, warmup $T_w = \min(\lfloor T/10 \rfloor, 200)$ for a budget of T steps, and gradients clipped to global norm 1. There is no weight decay, because the objective is a variational energy and any penalty on the parameters would bias it. Table 8 lists the settings shared by all runs.

Table 8: Settings shared by every GS-DFT run. Settings that vary by experiment are in Table 9.

Setting	Value	Reference
Optimizer	Adam, clip at global norm 1, no weight decay	above
Learning rate	10^{-2} peak, 10^{-4} floor, cosine	above
Screening	$\tau = 10^{-7}$ on the normalized overlap, see below	Section 4
Pair-list padding	2.0	Section C.4
Fit regularizer	$\lambda = 10^{-8}$	Proposition A.4
Eigensolve	$\varepsilon = \delta = 10^{-4}$	Equation (12)
Coulomb quadrature	20 Gauss–Legendre nodes	Section C.2
Precision	float64	Section C.1

Grids. The exchange–correlation quadrature uses the Becke grids of `dftax`, indexed by a level L that sets the number of radial shells and the Lebedev order per atom,

$$L = 2 : (50, 194), \qquad L = 3 : (75, 302), \qquad L = 5 : (110, 590).$$

Training uses level 3, except on the protein-scale systems, which use level 2. The observables of Figure 3 are evaluated on a common level-5 grid, for the Gaussian references and for GS-DFT alike, so that the density errors compare the same point set and a cloud cannot profit from the grid it was trained on.

Screening. Pair screening is applied to every run of Figure 3, and to the other experiments whenever $M \geq 300$.

Initialization. Two initializations are used, and Table 9 states which run uses which. Both place the cloud from the geometry and tabulated element constants alone, with no reference calculation for the molecule.

(i) *Atom-centered.* The M splats are divided evenly over the atoms, each atom's splats are isotropic, $\boldsymbol{A} = 2\alpha\mathbb{1}$, with Gaussian exponents α log-spaced between e^{-1} and $e^{4.5}$, and the centers and log-eigenvalues receive Gaussian perturbations of scale 0.1 and 0.02. The coefficients are drawn as $\boldsymbol{C}_{\mu k} \sim \mathcal{N}(0, 0.1^2)$.

(ii) *Chemically informed.* A quarter of the splats are placed at bond midpoints, with bonds detected from covalent radii, and the rest on the atoms in proportion to the nuclear charge, with at least one per atom. Atom-centered splats are isotropic, $\boldsymbol{A} = 2\alpha\mathbb{1}$, with Gaussian exponents α log-spaced over the range of the element's primitive exponents in def2-SVP. Bond-centered splats have precision eigenvalue $10/d^2$ across a bond of length d and $4/d^2$ along it, so they are elongated along the bond, and a bond that carries several splats spreads these values over a factor of 2. The same perturbations as in (i) break the remaining symmetry. The coefficients are then either random, as in (i), or the minimal-basis guess: the tabulated atomic occupations of each element are projected onto the splats, and the N_{occ} natural orbitals of largest occupation are kept.

The comparison between these choices is an ablation (Section E).

Per-experiment settings. Table 9 gives the settings that differ between experiments. The refresh period differs because the size ladder was tuned to $K = 10$ after the basis-accuracy ladders had been run at $K = 50$. The protein-scale runs have individual budgets: $4{,}500$ steps for Trp-cage, $1{,}500$ for defensin, $1{,}000$ for insulin, 972 for β-spectrin, which ran to its walltime limit, and 600 for 3IFU.

Table 9: Settings that vary by experiment. n_{ao} is the function count of the named Gaussian basis.

Result	Cloud size	Init	Steps	K	Grid	Seeds
Observables, Figure 3	ladder of M	(i)	12,000	10	3	3
Accuracy per parameter, Figure 5	ladder of M	(i)	12,000	50	3	3
Functional coverage, Table 1	$M = n_{\mathrm{ao}}$(cc-pVTZ)	(i)	48,000	50	3	3
Ionization potentials, Table 2	$M = n_{\mathrm{ao}}$(def2-QZVPP)	(i)	12,000	50	3	3
Anion convergence, Figure 4(a)	ladder of M	(i)	12,000	50	3	3
LiF dissociation, Figure 4(b)	$M = n_{\mathrm{ao}}$(cc-pVTZ)	(i)	4,000	50	3	3
Protein-scale systems, Table 3	$M = n_{\mathrm{ao}}$(cc-pVTZ)	(ii), minimal-basis $\boldsymbol{C}$	per system	10	2	1

C.6 Evaluation protocol

Reported energies. Every reported GS-DFT energy on the molecules of Figures 3 and 5 and Tables 1 and 2 is evaluated after training with the exact Hartree energy of case (iii) in Section C.2, over the full pair list and without screening. For these runs, the density-fitting error of Section A.3 affects the trained parameters but not the reported energy of those parameters, which is variational for semilocal functionals. The protein-scale energies of Table 3 are the fitted training objective, because the $\mathcal{O}(M^4)$ exact evaluation is out of reach at $M \approx 2 \times 10^4$, and they carry the density-fitting error of Section A.3. For hybrid functionals and Hartree–Fock, the exchange term is evaluated with the fit on the converged auxiliary basis, because the exact four-index exchange tensor does not fit in memory at the cloud sizes where the comparison is made.

Ionization potentials. Direct minimization returns an occupied subspace, not canonical orbitals. The orbital energies are recovered by one Fock build in the converged basis, followed by an $N_{\mathrm{occ}} \times N_{\mathrm{occ}}$ eigensolve in the occupied subspace, and the ionization potential is $-\varepsilon_{\mathrm{HOMO}}$ under Hartree–Fock (Koopmans, 1934). No further self-consistent iteration is run, since it would report a nearby stationary point instead of the minimized state.

Forces. Forces are the explicit partial derivative of Proposition A.3, evaluated by one reverse pass in $\boldsymbol{R}$. By Equation (16), their error is linear in the optimization residual, so checkpoints used for forces are first polished. The polish continues training in three legs of 2000 steps at learning rates 10^{-3}, $3 \cdot 10^{-4}$ and 10^{-4}, then solves for $\boldsymbol{C}$ by a self-consistent-field iteration in the frozen cloud,

accepted only if it lowers the energy, and finishes with 1000 steps at 10^{-4}. The protein-scale forces of Table 3 are the explicit partial derivative at the training checkpoint, without this polish, reported as the root mean square over all Cartesian components,

$$F_{\mathrm{rms}} = \Big(\frac{1}{3N_{\mathrm{at}}} \sum_{a,\,i} F_{a,i}^2 \Big)^{1/2}.$$

Limits. Two estimators of a basis-set limit are used. The Gaussian limit of Figure 3 and Table 1 is the three-point geometric extrapolation of the last three rungs E_1, E_2, E_3,

$$r = \frac{E_3 - E_2}{E_2 - E_1},$$
$$E_{\mathrm{CBS}} = E_1 + \frac{E_2 - E_1}{1 - r},$$

which is used when $0 < r < 1$; otherwise the largest rung is kept. Applied pointwise to densities and forces, the same rule gives the field references of the second row of Figure 3. The limits of Figure 5 come from a power-law fit over all rungs of each ladder,

$$E_P - E_\infty \propto P^{-\alpha},$$

with E_∞ constrained below every rung and chosen to maximize the coefficient of determination of the linear fit of $\log(E - E_\infty)$ against $\log P$. For each E_∞ the inner fit is linear and exact, so the estimator is a one-dimensional search.

Observables. For a density ρ and a reference ρ_{ref} with N electrons, both evaluated on the same grid, the density error is the total variation per electron,

$$\mathrm{TV} = \frac{1}{2N} \sum_g w_g \left| \rho(\boldsymbol{r}_g) - \rho_{\mathrm{ref}}(\boldsymbol{r}_g) \right|.$$

The force error is the largest absolute Cartesian component of the difference,

$$\Delta F = \max_{a,\,i} \left| F_{a,i} - F_{a,i}^{\mathrm{ref}} \right|.$$

Energy errors in Figure 3 are reported per atom, so that systems of different size share an axis.

References. Gaussian reference energies, densities and forces are computed with `dftax` on the same training grid as the GS-DFT run they are compared with, and the reference observables of Figure 3 are evaluated on the same level-5 grid as the GS-DFT ones. Alanine-dipeptide force references and the quadruple- and quintuple-zeta ethanol references are computed with PySCF, and agree with `dftax` at cc-pVTZ to within 0.1 mHa on ethanol for every functional used and to within 0.25 mHa on the dipeptide. Matched comparisons set $M = n_{\mathrm{ao}}$ of the named basis, so that both representations have the same number of functions and, by Equation (32), nearly the same number of parameters.

C.7 Reproducibility

Hardware. Runs used NVIDIA A100 and H100 GPUs with 80 GB of memory, and H200 GPUs with 141 GB for the largest protein and for the comparison of Section D. Multi-GPU runs used four GPUs of one node.

Software. Python 3.13, JAX 0.10 with CUDA 12, Equinox 0.13, Optax 0.2 and `dftax` 0.8. Versions are pinned in the lockfile of the repository.

Geometries. Every molecular geometry used in Section 5 is provided in the repository, except the protein structures of Table 3, which are available from the FMODB database (Takaya et al., 2021).

Code layout. The repository separates the method from the experiments. The package holds the chart, the integrals, the density fit, screening, the eigensolve, sharding and the trainer, and each experiment of Section 5 has its own directory with a runner, a configuration file, a launcher and the plotting script that produces its figure or table. Every run prints one machine-readable result line with its settings and outputs, and the figures are built from those lines. The configuration of every reported run is recorded in its result line, so any row of a table can be regenerated from its settings alone.

D Comparison with differentiable DFT frameworks

We compare GS-DFT with the three open-source differentiable DFT codes: D4FT (Li et al., 2023) and MESS (Helal & Fitzgibbon, 2024), both written in JAX, and DQC (Kasim et al., 2022), written in PyTorch. PySCF (Sun et al., 2020) is not differentiable and enters only as the reference energy every code is measured against. All codes solve restricted Kohn–Sham DFT with the PBE functional on the same geometries.

D.1 Capabilities

Table 10 lists what each code supports. A code is marked as supporting relevant bases when it reproduces the PySCF energy of water to within 0.1 mHa in both cc-pVDZ and cc-pVTZ, the bases used in production chemistry (Section D.3). MESS passes this test at STO-3G and 6-31G, and fails from cc-pVDZ upwards. D4FT passes it in double precision, while its default single-precision mode returns NaN. DQC runs on the CPU only, because its interface to the integral library converts GPU tensors to host arrays and fails when the molecule is placed on a GPU. Among the differentiable codes, GS-DFT is the only one that distributes over several GPUs, and none of the others offers both density fitting and forces on a GPU. The largest system that another differentiable code completes at cc-pVTZ is ethanol, with 9 atoms, in DQC.

Table 10: Capabilities of differentiable DFT codes, with PySCF as the non-differentiable reference. Relevant bases means agreement with PySCF to within 0.1 mHa for water in cc-pVDZ and cc-pVTZ. The last column is the largest system, in atoms, completed at cc-pVTZ on one node before OOM.

Code	GPU	Multi-GPU	Differentiable	DF	Forces	Relevant bases	Largest system
GS-DFT (ours)	✓	✓	✓	✓	✓	✓	2,742
D4FT (Li et al., 2023)	✓	✗	✓	✗	✗	✓	9
MESS (Helal & Fitzgibbon, 2024)	✓	✗	✓	✗	✗	✗	–
DQC (Kasim et al., 2022)	✗[a]	✗	✓	✓	✓	✓	9
PySCF (Sun et al., 2020)	✗	✗	✗	✓	✓	✓	153[b]

[a] The GPU path fails while building the molecule. The last column is the largest system, in atoms, that each code completes at cc-pVTZ on one node before OOM; a dash marks a code that fails at cc-pVTZ on water. [b] PySCF's entry is set by wall-clock time on the CPU, not by memory.

D.2 Protocol

We test each code in two steps. The first is a correctness check on water in the cc-pVDZ basis, where every code should agree with PySCF. The second is a size ladder in the cc-pVTZ basis: water, ethanol, alanine dipeptide, and alanine chains of 5, 15 and 45 residues, followed by insulin in a 784-atom structure, not the 947-atom FMODB entry of Table 3. GS-DFT runs with as many splats as the cc-pVTZ basis has functions, $M = n_{\mathrm{ao}}$. Every code uses the PBE functional, the level-3 PySCF grid or the closest grid the code exposes, and its own default optimizer and convergence criterion. Where a code has no grid level of its own, its grid is built by PySCF at level 3, as MESS does by default. Every code runs on one node with no time limit and climbs the ladder until its first out-of-memory error or crash. The GPU codes use one node with eight H200 GPUs of 141 GB each: GS-DFT uses four, as in Table 3, and D4FT and MESS use one, since neither distributes over several. PySCF and DQC run on the CPU with all 64 cores and 500 GB of a CPU node. We record the energy, the peak GPU memory, the number of iterations, and the total time.

D.3 Agreement on a small molecule

Table 11 reports the water check. D4FT in double precision agrees with PySCF to 0.04 mHa at cc-pVDZ and to 0.05 mHa at cc-pVTZ, the level of grid differences between codes, while its default single-precision mode returns NaN after the full $4{,}000$-step budget. MESS uses Cartesian basis functions, so we compare it against PySCF in the same Cartesian basis. It agrees to 0.006 mHa at STO-3G and at 6-31G, and then fails once the basis carries polarization functions. At cc-pVDZ it returns an energy 589 mHa *below* the reference, which no variational method can do, and with exact exchange it returns -77.836 Ha against a Hartree–Fock energy of -76.027 Ha. At cc-pVTZ it fails in its integral callback before the first iteration. The failure therefore arrives with the d shells of cc-pVDZ and not with the s and p shells of the smaller bases, which points at the integrals rather than at the optimizer. GS-DFT at a matched count of $M = n_{\mathrm{ao}} = 24$ ends 66.6 mHa above the cc-pVDZ reference. This is the smallest rung in this work, and it lies below the crossing of the two scaling laws of Figure 5: the splat and Gaussian errors fall with different exponents, so the fixed basis stays ahead until the parameter count passes the crossing point.

Table 11: Water, cc-pVDZ, PBE, level-3 grid. Δ is the energy minus the PySCF energy. Times are wall-clock seconds including compilation. Iterations are SCF cycles for PySCF and optimization steps for the others. [a]Cartesian basis, compared against Cartesian PySCF (-76.33470 Ha).

Code	E (Ha)	Δ (mHa)	**Iterations**	**Time (s)**
PySCF	-76.33344	–	7	1.9
D4FT (float64)	-76.33340	$+0.038$	637	128
D4FT (float32, default)	NaN	–	4,000	206
MESS[a]	-76.92339	-588.7	110	9.7
DQC (CPU)	-76.33345	-0.010	–	15.4
GS-DFT ($M = 24$)	-76.26686	$+66.6$	12,000	154

D.4 Scaling with system size

Table 12 reports the cc-pVTZ ladder. The differentiable baselines stop early. MESS fails on water before its first iteration. DQC reaches ethanol and then runs out of host memory on alanine dipeptide, because in its default mode it stores the four-index electron repulsion tensor, which for $n_{\mathrm{ao}} = 468$ is 384 GB in double precision. D4FT reaches ethanol only by filling 138.9 of the card's 141 GB, and on alanine dipeptide its own estimate for the integral precomputation alone is eight hours. Every differentiable baseline therefore stops at ethanol or before, while GS-DFT runs the whole ladder. GS-DFT climbs to insulin at $17{,}448$ functions in 18.4 GB, and its peak memory grows by a factor of 102 across the ladder while the basis grows by a factor of 300.

GS-DFT is slower in wall-clock time than a converged SCF on these systems, because it optimizes the basis together with the orbitals over $12{,}000$ steps where an SCF needs tens of cycles at a fixed basis. Its cost per step grows from 0.10 s at water to 80 s at insulin, a factor of 800 across a 300-fold increase in basis size. What it offers in exchange is reach: the same method runs the $2{,}742$-atom FMO systems of Table 3 on one node, while every other differentiable code here stops at ethanol or before.

E Ablations

Each subsection varies one setting of Section C.5 and holds the rest at their production values. Energies are totals in Hartree and differences are in mHa.

E.1 Regularized eigensolve

The regularized eigensolve of Proposition A.6 keeps the eigendecomposition gradient finite when the occupied Gram matrix has degenerate eigenvalues, where the standard gradient divides by zero. We train four symmetric molecules for $12{,}000$ steps with it and without it (Table 13). Without the regularization, methane and benzene stall more than 40 mHa above the regularized result. On CO_2 and N_2 the two agree to within 2 mHa, as they do on a 45-residue alanine chain at matched wall-clock time. The regularization removes a failure on some symmetric molecules and costs nothing elsewhere.

Table 12: The cc-pVTZ ladder on one node (Section D.2). Memory is the peak GPU memory in GB. GS-DFT reports time per optimization step and PySCF time per SCF cycle, both in seconds; D4FT and DQC report the wall-clock time of the whole run, since neither exposes a per-iteration cost. GS-DFT runs on four H200 GPUs at every rung. GS-DFT uses $M = n_{\mathrm{ao}}$ splats; its time per step is the median over 100 steps after compilation. OOM marks an out-of-memory failure, and a dash marks a system not attempted after an earlier failure or, for PySCF, one that did not finish within the day of wall-clock time it was given.

			GS-DFT		D4FT	DQC	PySCF
System	**Atoms**	n_{ao}	Mem	Time/step	Wall (s)	Wall (s)	Time/cycle
Water	3	58	0.18	0.10	474	19.9	0.7
Ethanol	9	174	1.37	0.11	4,365	29.3	1.2
$(\mathrm{Ala})_2$	22	468	1.49	0.15	–	OOM	4.6
$(\mathrm{Ala})_5$	53	1,158	1.76	0.34	–	–	22.9
$(\mathrm{Ala})_{15}$	153	3,358	6.43	2.06	–	–	238
$(\mathrm{Ala})_{45}$	453	9,958	9.45	21.0	–	–	–
Insulin	784	17,448	18.43	80.1	–	–	–

Which symmetries trigger it stays open, since CO_2 and N_2 also have degenerate occupied orbitals and do not stall.

Table 13: Regularized eigensolve against a plain eigendecomposition, at matched settings. Δ is the energy without the regularization minus the energy with it.

System	M	**Regularized**	**Plain**	Δ (mHa)
CH_4	86	-40.51841	-40.47054	$+47.9$
C_6H_6	264	-232.09545	-232.05306	$+42.4$
CO_2	90	-188.48252	-188.48038	$+2.1$
N_2	60	-109.45544	-109.45653	-1.1

E.2 Optimizer

GS-DFT trains with plain Adam and no weight decay. We compare every first-order optimizer in Optax on alanine dipeptide at $M = 468$. Each one takes its peak learning rate from a 600-step exponential range test and then runs the production schedule of Section C.5 for 12,000 steps (Table 14). The optimizers split into three groups. Adam, five of its variants and LAMB finish within 14 mHa of Adam, the best being Adamax at 13.7 mHa below it. Yogi, RMSProp and AMSGrad finish 75 to 94 mHa higher. The rest finish 0.2 to 1.0 Ha higher, and Lion and Adan diverge.

Table 14: Optimizers on $(\mathrm{Ala})_2$ with $M = 468$. Δ is the final energy minus that of Adam.

Optimizer	**Family**	Δ (mHa)	**Optimizer**	**Family**	Δ (mHa)
Adamax	Adam	-13.7	Novograd	Adam	$+228.6$
AdaBelief	Adam	-9.5	AdaGrad	second moment	$+302.2$
RAdam	Adam	-9.4	signSGD	sign	$+321.3$
LAMB	trust ratio	-8.3	SGD	non-adaptive	$+449.3$
AdamW	weight decay	-4.2	LARS	trust ratio	$+847.2$
Adam	Adam	0.0	Fromage	trust ratio	$+998.8$
NAdam	Adam	$+5.4$	AdaDelta	second moment	$+1043.4$
Yogi	Adam	$+75.0$	Lion	sign	diverged
RMSProp	second moment	$+88.9$	Adan	Adam	diverged
AMSGrad	Adam	$+94.4$			

E.3 Initialization

The chemically informed initialization of Section C.5 has four ingredients, which we add one at a time to a random atom-centered start: element exponent ranges from def2-SVP (S1), bond-centered

splats (S2), bond-aligned anisotropy (S3), and splats allocated proportionally to nuclear charge (S4). Table 15 reports water at $M = 48$ and ethanol at $M = 144$ after 4,000 steps. The full initialization ends 27 mHa lower on water and 35 mHa lower on ethanol than random placement, with S1 and S4 contributing most. The seed spread falls from 11.7 to 2.1 mHa on water and from 4.6 to 0.1 mHa on ethanol, so one run is representative.

Table 15: Initialization ingredients, added cumulatively. Median energy over seeds after 4,000 steps, with the seed spread in mHa.

		H_2O, $M = 48$		C_2H_5OH, $M = 144$	
	Adds	**Energy**	**Spread**	**Energy**	**Spread**
S0	random, atom-centered	−76.30164	11.7	−154.87106	4.6
S1	element exponent ranges	−76.32034	1.3	−154.89223	1.9
S2	bond-centered splats	−76.32044	1.3	−154.89412	6.0
S3	bond-aligned anisotropy	−76.32078	2.3	−154.89656	1.7
S4	charge-proportional allocation	−76.32853	2.1	−154.90574	0.1

E.4 Refresh period

The pair list and the auxiliary basis are rebuilt from the current splats every K steps and frozen in between (Section C.5). A small K follows the moving cloud closely but pays for rebuilds often. We vary K alone on alanine dipeptide at $M = 468$ for 12,000 steps at $\tau = 10^{-7}$ (Table 16). Cost falls monotonically with K and saturates early: rebuilding every step costs 76% more than rebuilding every 200 steps, and $K = 10$ is already within 8% of the cheapest setting. The energies span 86 mHa without ordering in K, so at this budget K sets cost and not accuracy. We use $K = 10$.

Table 16: Refresh period K on alanine dipeptide, $M = 468$, 12,000 steps, $\tau = 10^{-7}$. Time is the total wall-clock time of the run on one A100.

K	**Energy** (Ha)	**Time** (s)	**ms / step**
1	−495.60818	9,629	802
5	−495.59166	6,427	536
10	−495.63820	5,878	490
50	−495.57240	5,580	465
200	−495.55246	5,462	455

E.5 Screening threshold

A density pair enters the Coulomb term only if its product has weight above τ (Section C.4), which fixes both the cost of a step and the error in the Coulomb energy. We repeat the run above at $K = 10$ and vary τ, where $\tau = 0$ keeps every pair (Table 17). At $\tau = 10^{-8}$ screening already removes half of the pairs and 46% of the wall-clock time, and four further decades remove only 12% more. The energies again span 63 mHa without ordering in τ, the same scale as the refresh sweep. We use $\tau = 10^{-7}$, at the knee of the pair count.

Table 17: Screening threshold τ on alanine dipeptide, $M = 468$, 12,000 steps, $K = 10$. Pairs is the number retained at the last refresh, out of 203,560 unscreened.

τ	**Energy** (Ha)	**Pairs**	**Retained**	**Time** (s)
0	−495.61730	203,560	100%	11,025
10^{-8}	−495.59341	102,450	50%	6,310
10^{-7}	−495.63820	94,658	47%	5,878
10^{-6}	−495.65585	86,488	42%	5,503
10^{-5}	−495.61626	76,734	38%	5,019

F ADDITIONAL RESULTS

F.1 BARE ANIONS

Figure 4(a) shows the fluoride ion. Table 18 extends it to the hydroxide ion and gives the full ladder: for each cardinal number ζ, a cloud with the function count of the plain cc-pVζZ basis, trained with the atom-centered initialization of Section C.5, against the plain and the augmented Gaussian bases of the same cardinal number. The cloud receives no diffuse functions and no information about the anion beyond the nuclei and the electron count.

Table 18: Bare anions at matched function count. Energies in Hartree, restricted PBE. M is the function count of the plain cc-pVζZ basis. Bold marks a splat energy below the augmented basis of the same cardinal number.

	ζ	M	cc-pVζZ	aug-cc-pVζZ	GS-DFT	Spread
F^-	D	14	−99.66562	−99.77465	−99.08264	304.1
	T	30	−99.74783	−99.79967	**−99.80099**	4.7
	Q	55	−99.77711	−99.80743	**−99.81283**	0.3
	5	91	−99.79765	−99.81012	**−99.81330**	0.1
OH^-	D	19	−75.63289	−75.73520	−75.54983	83.2
	T	44	−75.70329	−75.75426	**−75.75548**	1.8
	Q	85	−75.72893	−75.76038	**−75.76674**	4.4
	5	146	−75.74814	−75.76270	**−75.77348**	0.2

From TZ onwards, the cloud lies below the augmented basis of the same cardinal number for both anions, while using only the functions of the plain basis. At QZ, it already lies below aug-cc-pV5Z, the largest augmented basis we compute. The diffuse support that augmentation adds by hand is therefore found by the optimizer, which moves splats outward from the nuclei.

F.2 DISSOCIATION ENERGY LADDERS

Figure 6 gives the total-energy dissociation curves $E(R)$ of LiH and LiF. Each panel compares the plain cc-pVζZ bases and their augmented aug-cc-pVζZ counterparts, for $\zeta = \mathrm{D}, \mathrm{T}, \mathrm{Q}$, against splat clouds with the function counts of the plain bases. Restricted PBE keeps both electrons of the bond paired, so as R grows the closed-shell solution follows the ionic diabat Li^+X^- rather than dissociating into neutral atoms. This is the state every curve in Figure 6 is meant to describe, and at large R it needs the diffuse density of an isolated anion, which is where the plain bases fail.

The Gaussian references are single-point restricted Kohn-Sham calculations from an atomic guess at each R. At the most stretched geometries, the larger bases converge to a higher self-consistent solution instead of the ionic ground state. We identify these points variationally: a larger basis cannot give a higher energy for the same state, so a geometry is excluded from the matched-count margins of Section 5.2 wherever the augmented ladder is not non-increasing in ζ,

$$E_{\text{aug-cc-pVDZ}}(R) \;\geq\; E_{\text{aug-cc-pVTZ}}(R) \;\geq\; E_{\text{aug-cc-pVQZ}}(R).$$

At LiH, $R = 6$ Å, aug-cc-pVQZ lies 335 mHa above aug-cc-pVDZ, which is a convergence failure of the baseline rather than a property of the representation. These points are shown in Figure 6 but not used in any comparison. On the other hand, the splat curves are computed as one continuation along R rather than as independent calculations. The first geometry starts from the atom-centered initialization, while each following geometry starts from the converged cloud of the previous one: splats closer to the displaced atom than to Li are translated with it, the coefficients are kept. The cloud therefore follows the density from the covalent to the ionic regime instead of restarting at every geometry. Each point is the lower of the continued and an independent calculation at that R. The continued clouds track the ionic diabat below the entire ladder at matched function count.

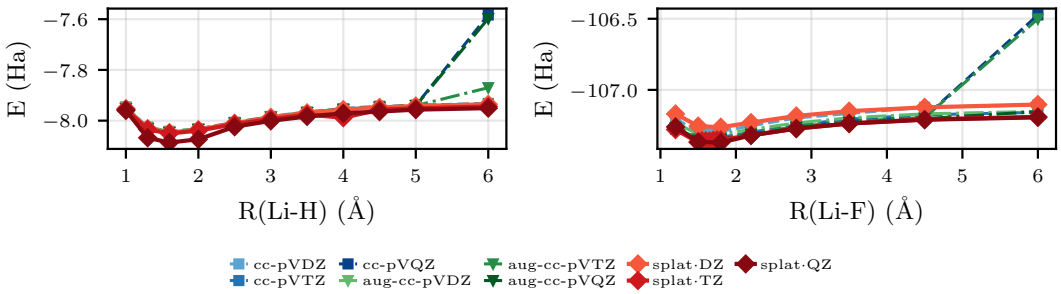

Figure 6: Dissociation energy ladders $E(R)$ for LiH (left) and LiF (right), restricted PBE on the ionic Li^+H^-/Li^+F^- diabat. Plain cc-pVDZ/TZ/QZ (blue), augmented aug-cc-pVXZ (green), and splats sized to the plain cc-pVXZ counts (red).