WILEY QUANTUM CHEMISTRY
International Journal of

Numerical methods for the inverse problem of density functional theory

Daniel S. Jensen[1] (iD) | **Adam Wasserman[2,3]**

[1]Center for Computing Research MS 1321, Sandia National Laboratories, Albuquerque, New Mexico, 87185-1321

[2]Department of Chemistry, Purdue University, West Lafayette, Indiana 47907

[3]Department of Physics and Astronomy, Purdue University, West Lafayette, Indiana 47907

Correspondence
Adam Wasserman, Department of Chemistry, Purdue University, West Lafayette, IN 47907.
Email: awasser@purdue.edu

Abstract

The inverse problem of Kohn–Sham density functional theory (DFT) is often solved in an effort to benchmark and design approximate exchange-correlation potentials. The forward and inverse problems of DFT rely on the same equations but the numerical methods for solving each problem are substantially different. We examine both problems in this tutorial with a special emphasis on the algorithms and error analysis needed for solving the inverse problem. Two inversion methods based on partial differential equation constrained optimization and constrained variational ideas are introduced. We compare and contrast several different inversion methods applied to one-dimensional finite and periodic model systems.

KEYWORDS

density functional theory, inverse problems, PDE-constrained optimization

1 | INTRODUCTION

Density functional theory (DFT) is a very popular and formally exact quantum many-body theory that has successfully been used to model both finite and periodic systems.[1,2] Much of this success stems from the availability of approximate exchange-correlation functionals used as input in the direct problem of DFT to produce the ground-state electronic density of a given system. Although far less common, this process can be reversed and the true exchange-correlation potential can be recovered from a given electronic density by solving the inverse problem of Kohn–Sham (KS) DFT. (Other flavors of DFT exist but we focus on the standard ground-state version of KS DFT and drop the KS prefix for brevity throughout this tutorial.) Although both the forward and inverse problems use the same equations, the difference in input and output variables leads to major differences in solution methods for the two problems. We refer to the inverse problem of DFT as a density-to-potential inversion and the numerical methods required for solving this problem are the main focus of this tutorial. In this section, we give a brief introduction to inverse problems and explain how density-to-potential inversions can be used within DFT.

Inverse problems are common in science and have been central in quantum mechanics since its inception. Much of what we know about the structure of matter has come from scattering experiments,[3] which can be described mathematically as inverse problems. A variety of mathematical and numerical techniques exist for solving scattering problems as well as other inverse problems.[4] These methods are often very different from the methods used for solving the direct problems due to the differences in mathematical structure and input data.

According to Hadamard,[5] a problem is well-posed if a solution exists, it is unique, and it depends continuously on the data. If any of the three properties listed above is violated then the problem is ill-posed. The forward or direct problem of DFT is widely considered to be well-posed thanks to several proofs including the Hohenberg–Kohn Theorems and the constrained-search formulation of Levy and Lieb.[1] The inverse problem of DFT can also be well-posed for discretized systems as proven in Ref. [6] but often errors and missing information in the input density lead to violations of Hadamard's well-posed conditions. In such cases, special precautions must be taken in designing inversion methods that converge to the true solution of the inverse problem.

We have found that understanding the numerical errors involved in the inverse problem of DFT is extremely helpful in developing successful DFT inversion methods. Appendix A and subsection 2.2.1 contain a number of examples showing how the numerical error in both the inversion algorithm and the target density being inverted set bounds on the inverted potential's accuracy. A central theme in this tutorial is the creation of robust inversion methods that use information about the numerical errors involved to produce accurate potentials. This information can be used to properly smooth target densities or choose a form of regularization. Regularization is the general term used to describe a method for solving ill-posed problems. Some regularization methods are as simple as stopping an optimization according to an a posteriori parameter choice rule while others, such as total variation, can be sophisticated and very challenging mathematical and programming exercises.[12] Often the form of regularization is chosen to penalize features known to be incorrect based on a priori information about the solution. For a thorough review of regularization and inverse problem theory see Refs. [7,12,13]. The density-to-potential inversion methods presented in this tutorial are designed explicitly to include regularization.

We make a special effort to verify the integrity and robustness of the inversion algorithms presented here. Our test cases are all noninteracting systems because we can focus on recovering only the known external potential as explained in section 3. These systems may seem trivial but their corresponding inverse problems are just as difficult as the inverse problems corresponding to interacting densities and provide great insight into the design of inversion algorithms. We also strive to use different algorithms and grids for the direct problems in the examples with approximate target densities to avoid any reliance on inverse crimes as described in Appendix A (Boundary conditions and inverse crimes). The numerical errors in both the inversion methods and the target densities used as input are also tracked carefully to avoid problems with overfitting. In general, a great deal of numerical analysis is used throughout this work to make sure that our inversion methods are both correct and efficient.

One possible motivation for studying the inverse problem of DFT is to benchmark the many approximate exchange-correlation functionals currently available. Benchmarking is usually done on model systems where the density is either known analytically or found numerically through a direct solution of the electronic Schrödinger equation. As seen in many works on inverse problems in DFT,[14–17] major features of the exact exchange-correlation potential are often found to be missing in popular approximations and, thus, the inverse problem can help guide the development of new approximations.

Another motivation for studying the inverse problem of DFT is to calculate accurate non-additive kinetic potentials in the context of density-dependent embedding methods, as reviewed very recently by Banafsheh and Wesolowski.[18] Such inversions are also central to the development and application of Partition-DFT.[19]

Perhaps an even more practical reason to study the inverse problem of DFT comes from its connection with the optimized effective potential (OEP) method. In the OEP method, the exchange-correlation functional is an explicit functional of the KS orbitals and, therefore, an implicit functional of the density.[20] As pointed out in Ref. [17], the optimization procedures used in the OEP method can be adapted to perform density-to-potential inversions and vice versa. This connection with the OEP method shows that density-to-potential inversions are also important predictive tools in addition to their use in benchmarking or guiding the construction of potentials.

2 | DIRECT AND INVERSE METHODS

We begin our tutorial by highlighting the differences between the forward and inverse problems of DFT. Although the direct and inverse problems of DFT share the same KS equations, there are several important differences between them that lead to very different algorithms and convergence criteria. We briefly describe common numerical methods for solving the direct problem of DFT and then describe in detail several inverse-problem methods. All of the examples presented are one-dimensional systems for simplicity but the formulas are written for multidimensional systems to aid future development. We also limit our studies to spin-compensated systems and drop the spin index on most formulas.

The KS equations for a closed-shell system with $2N$ electrons can be written as

$$\varepsilon_j \phi_j\left(\mathbf{r}\right) = \left[-\frac{\Delta}{2} + v_{\mathrm{KS}}\left(\left[n\right], \mathbf{r}\right)\right]\phi_j\left(\mathbf{r}\right), \tag{1a}$$

$$n\left(\mathbf{r}\right) = 2\sum_{j=1}^{N}\left|\phi_j\left(\mathbf{r}\right)\right|^2, \tag{1b}$$

$$v_{\mathrm{KS}}\left(\left[n\right], \mathbf{r}\right) = v_{\mathrm{ext}}\left(\mathbf{r}\right) + v_{\mathrm{H}}\left(\left[n\right], \mathbf{r}\right) + v_{\mathrm{xc}}\left(\left[n\right], \mathbf{r}\right), \tag{1c}$$

where the underlined quantities are unknowns, $v_{\mathrm{xc}} = \frac{\delta E_{\mathrm{xc}}[n]}{\delta n}$ is the exchange-correlation potential,

$$v_{\mathrm{H}}\left([n], \mathbf{r}\right) = \int d\mathbf{r}'\, n\left(\mathbf{r}'\right)\frac{1}{|\mathbf{r} - \mathbf{r}'|} \tag{2}$$

is the Hartree potential, the external potential $v_{\mathrm{ext}}\left(\mathbf{r}\right)$ is usually the electron-nuclei potential, and the orbitals are constrained to be orthonormal:

$$\langle \phi_j | \phi_k \rangle = \delta_{j,k} = \begin{cases} 1 & \text{if } j = k \\ 0 & \text{otherwise} \end{cases}. \tag{3}$$

The orbitals are usually chosen such that their corresponding eigenvalues $\{\varepsilon_j\}$ form the set of lowest possible eigenvalues. The KS equations for the inverse problem of DFT

$$\varepsilon_j \underline{\phi_j(\mathbf{r})} = \left[-\frac{\Delta}{2} + v_{KS} \ ([n], \mathbf{r}) \right] \underline{\phi_j(\mathbf{r})}, \tag{4a}$$

$$n(\mathbf{r}) = 2 \sum_{j=1}^{N} \underline{|\phi_j(\mathbf{r})|^2}, \tag{4b}$$

$$v_{KS}([n], \mathbf{r}) = v_{ext}(\mathbf{r}) + v_H([n], \mathbf{r}) + \underline{v_{xc}([n], \mathbf{r})}, \tag{4c}$$

only differ from the direct problem in the unknowns as indicated by underlining.

Equation 1a is a nonlinear eigenvalue problem due to the KS potential's density dependence whereas Equation 4a is a linear eigenvalue problem. The inverse problem is still nonlinear but the nonlinearity comes from the definition of the density shown in Equation 4b. This subtle difference in nonlinearity means that the mixing schemes prevalent in numerical methods for the direct problem are not needed in solving the eigenvalue problem in the inverse problem. Additionally, the definition of the density in Equation 4b puts constraints on the values of the KS orbitals when solving the inverse problem that are not present in the direct problem. These additional constraints are exploited in the inversion method described in subsection 2.2.4. Finally, the nonlinearity in both problems makes the choice of initial guess very important as convergence is not guaranteed in many numerical methods when poor initial guesses are used.[21]

The KS equations apply to both finite and extended systems provided that the appropriate boundary conditions are applied. In the direct problem of DFT the orbitals are usually constrained to satisfy periodic or box-type (Dirichlet) boundary conditions as dictated by the system being studied.[22] In the inverse problem the boundary conditions may not be known exactly but the density constraint uniquely determines the orbitals' behavior at the boundaries. If boundary conditions are imposed then it is important that they agree with the density constraint as explained in Appendix A (Boundary conditions and inverse crimes). We show in this tutorial the alternative of letting the inversion method enforce the orbital boundary conditions to agree with the density constraint. This method has the advantage that no detailed knowledge of the boundary conditions is needed and the disadvantage that the values of the potential at the boundaries may need to be discarded after the inversion if the density at the boundaries contains some inherent error. We call this method density-constrained boundary conditions (DCBC) and discuss it further in subsection 2.2.1.

As mentioned in section 1, uniqueness is a key ingredient in solving both direct and inverse problems. Although the density in DFT is a unique quantity in both the forward and inverse problems, the KS orbitals that it is built from are not unique. As seen in Equation 1b, the density is formed from the absolute value squared of the orbitals and therefore the orbitals can only be unique up to a phase factor. Furthermore, any unitary transformation of the KS orbitals will also produce the same density. The KS potential in Equation 1c is also a source of nonuniqueness as it is only unique up to a constant.[6] As seen in subsection 2.2, most density-to-potential inversion methods force the unknown potential to be unique by imposing restrictions of some form.

Numerical convergence in the direct problem of DFT involves both a convergence in the iterations to self-consistency and a convergence for the eigenvalue problem solved at each iteration. This convergence is described in great detail in Ref. [21] and we only point out here that the potentials are usually given by analytic formulas. This means that the KS equations can be solved self-consistently and the error in the resulting density will depend mainly on the level of discretization employed and the choice of approximate exchange-correlation functional. Numerical convergence in the inverse problem of DFT is also limited by the chosen discretization method but it is the quality of the target density that sets a somewhat surprising limit on convergence as explained in this tutorial.

Although most of the test cases presented in this work have analytic formulas for the target densities, most target densities of real interest are not known analytically. If the numerical error present in a given target density is not accounted for properly in a density-to-potential inversion then overfitting becomes a problem and unphysical features develop in the recovered potential. Examples of overfitting and convergence limits related to the relative error of the target density are given in Appendix A.

2.1 | Direct-problem methods

A variety of numerical methods for solving the KS equations exist and can be placed in essentially three different categories: plane waves and grid methods, localized atomic-(like) orbitals, and atomic sphere methods.[1] We use the finite difference grid method for our calculations similar to the Octopus time-dependent density functional theory (TDDFT) code.[23] The finite difference method is capable of producing sharp features that would be difficult to find using other common electronic-structure basis sets. In this section, we review several aspects of the finite difference formulation of DFT that are necessary for solving the direct problem and also relevant to some of the inverse problem methods. A more thorough review of the finite difference method applied to DFT can be found in Ref. [24].

The orbitals, potentials, and density in the KS equations are represented as discrete points on a grid when using the finite difference method. The potentials are diagonal in the coordinate representation and the Laplacian is a very sparse matrix operator depending on the approximation used. An example fourth-order Laplacian for a regularly spaced grid with seven points is

$$\frac{1}{h^2}\begin{bmatrix} \frac{15}{4} & -\frac{77}{6} & \frac{107}{6} & -13 & \frac{61}{12} & -\frac{5}{6} & 0 \\ \frac{5}{6} & -\frac{5}{4} & -\frac{1}{3} & \frac{7}{6} & -\frac{1}{2} & \frac{1}{12} & 0 \\ -\frac{1}{12} & \frac{4}{3} & -\frac{5}{2} & \frac{4}{3} & -\frac{1}{12} & 0 & 0 \\ 0 & -\frac{1}{12} & \frac{4}{3} & -\frac{5}{2} & \frac{4}{3} & -\frac{1}{12} & 0 \\ 0 & 0 & -\frac{1}{12} & \frac{4}{3} & -\frac{5}{2} & \frac{4}{3} & -\frac{1}{12} \\ 0 & \frac{1}{12} & -\frac{1}{2} & \frac{7}{6} & -\frac{1}{3} & -\frac{5}{4} & \frac{5}{6} \\ 0 & -\frac{5}{6} & \frac{61}{12} & -13 & \frac{107}{6} & -\frac{77}{6} & \frac{15}{4} \end{bmatrix}, \tag{5}$$

where h is the grid spacing and sided finite differences are used at the boundaries. Different orders of approximate finite difference operators can be derived using the algorithm described in Ref. [25]. There are many possible boundary conditions and methods to implement them depending on the problem and accuracy desired.[26–28] Custom meshes can also be used that conform to the geometry of a given system as explained in Ref. [22]. One disadvantage of the finite difference method applied to DFT is the lack of a variational principle for most approximate finite difference operators. This means that the approximate energy may actually be smaller than the true energy found by solving the KS equations exactly. The approximate energy will approach the true energy as the grid is refined and/or the accuracy of the finite difference operator is improved but it does not have to approach it from above unless the finite difference operator is modified as in Ref. [29].

A variety of numerical algorithms are available for computing the potentials in Equation 1c. The Hartree potential v_H shown in Equation 2 satisfies the Poisson equation

$$\Delta v(\mathbf{r}) = -4\pi n(\mathbf{r}) \tag{6}$$

and has been studied extensively. The fast Fourier transform, fast multipole method, and conjugate gradient methods are just a few of the available algorithms for computing v_H; a thorough review of these methods can be found in Ref. [30]. As mentioned in section 1, there are many approximate exchange-correlation functionals and corresponding potentials. The Libxc library contains a wide selection of these approximations and is easily incorporated in DFT programs.[31] Exchange-correlation approximations can also be derived for systems with custom electron–electron interactions as shown in Refs. [8,16].

The KS equations are usually solved using sparse matrix eigenvalue solvers in contrast to the dense solvers typically used in basis-set methods.[24] The examples in this work use the `eigs` and `eig_banded` eigenvalue solvers in the SciPy library, which are interfaces to routines in the LAPACK and ARPACK libraries, respectively.[32] These solvers are capable of computing the eigenvalues and eigenvectors of a sparse matrix to machine precision and are very efficient when only a few eigenvalues and eigenvectors are needed. If the potentials used in a given calculation are exact and the solvers are allowed to converge to machine precision then the majority of the numerical error will come from the discretization of the Laplacian. An example of this error is shown in Figure 1 for the ground state density of the harmonic oscillator using a second-order approximate Laplacian. Although the majority of the error is in the central region, Figure 2 shows that the relative error is actually largest in the asymptotic regions. The approximate finite difference operators are very accurate in regions where the orbitals are well represented by polynomials and are less accurate in the asymptotic region where the orbitals are dominated by exponential decay. The large relative error is largely inconsequential for most direct DFT problems with almost no effect on the total energies but it does play a strong role in the inverse problem as shown in subsection 2.2. This example also shows that the errors in most direct DFT problems are systematic rather than random noise typical of experimental data and must also be accounted for in the inverse problem as explained in Appendix A.

The previous example of a noninteracting system only required solving one eigenvalue problem but a typical DFT calculation for an interacting system requires multiple eigenvalue solves in what is known as the self-consistent field (SCF) method. The initial density guess for a given system is inserted into the right-hand side of Equation 1a, the eigenvalue problem is solved for a new set of orbitals, the orbitals are inserted into Equation 1b to produce a new density, and the cycle repeats until a convergence criterion is reached. This method is usually modified to avoid oscillating densities by mixing the new density with previous densities via the direct inversion in the iterative subspace (DIIS) method[33] or some other form of mixing.[22] Often the SCF loop is stopped when the change in density or the change in each eigenvalue becomes smaller than a prescribed value but this does not guarantee global convergence. An alternative gradient-based method that avoids the repeated eigenvalue computation is also possible as illustrated in Ref. [34]. A more detailed analysis of numerical convergence in DFT calculations can be found in Ref. [21].

2.2 | Inverse-problem methods

As is the case with the direct problem, there exist an assortment of numerical methods for solving the inverse problem of DFT, each with its own set of advantages and disadvantages. Here, we review several density-to-potential algorithms and introduce two new inversion methods. All of the
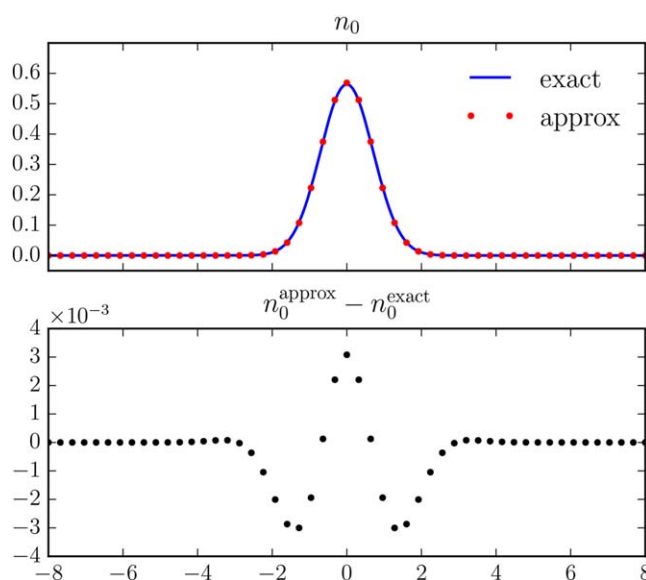
FIGURE 1 The ground state density of the harmonic oscillator compared to a numerical approximation using the eig_banded routine and a second-order finite difference approximation to the Laplacian (top). The majority of the error is in the high-density central region (bottom)

methods are presented using the finite-difference method and may include small adaptations made to convert algorithms that were originally designed for basis-set methods.

2.2.1 | One and two electrons: One-orbital formula

The one-orbital inversion formula

$$v_{KS}([n], \mathbf{r}) = \frac{\Delta \phi_0(\mathbf{r})}{2\phi_0(\mathbf{r})} = \frac{\Delta \sqrt{n_0(\mathbf{r})}}{2\sqrt{n_0(\mathbf{r})}} \tag{7}$$

can be derived simply by setting the energy to zero and solving for the KS potential in Equation 1 in terms of the ground state density n_0. (We are allowed to set the energy to zero because the potential is only unique up to a constant.) This formula is exact for one electron or two electrons with opposite spins in the same orbital and is often referred to as the bosonic or one-electron potential. (In the case of two electrons, the density $n_0(\mathbf{r})$ must be halved for the formulas in this subsection.)[35] Although at first glance the one-orbital formula appears somewhat trivial, it is helpful in
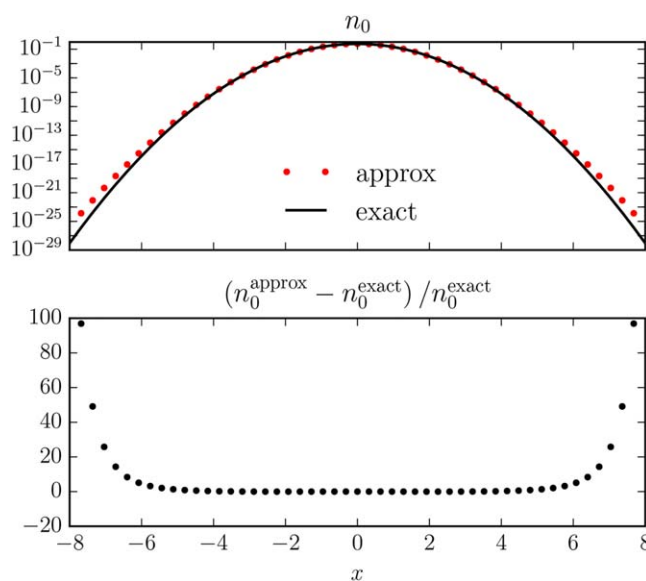

FIGURE 2 The same densities displayed in Figure 1 are displayed on a logarithmic scale (top). The relative error in the asymptotic regions is orders of magnitude larger than the relative error in the central region (bottom)

unraveling many of the numerical problems common to all density-to-potential algorithms and can even serve as a useful approximation for systems with many electrons.

The one-orbital formula makes it very clear that the KS orbital is uniquely determined up to an arbitrary phase by the target density via the equation $\phi_0 = \sqrt{n_0}$. This means that the values of the orbital in the entire domain of the problem are known including the boundaries. In the forward problem of DFT we usually force the orbitals to satisfy certain boundary conditions but in the inverse problem this is not necessary because the target density uniquely determines their behavior at the boundaries up to an arbitrary unitary transformation of the orbitals. Both Figure A1 and the g_j' ($\pm L$) ≈ 0 line in Figure 9 show examples of molecular system inversions with approximate boundary conditions enforced. In both examples the resulting potentials have several incorrect values near the boundaries where the potential actually enforces the approximate boundary conditions. In Figure A1, the potential should actually be

$$v(x) = \frac{x^2}{2}[\Theta(x-4) - \Theta(x+4)] - x[\delta(x-4) + \delta(x+4)] + [\delta'(x-4) - \delta'(x+4)]/2, \tag{8}$$

because the density is artificially forced to be zero at the boundaries and the "incorrect" potential values near the boundaries are approximations to this discontinuous behavior, where $\delta(x)$ is the Dirac delta function and $\Theta(x)$ is the Heaviside function. When approximate boundary conditions are not enforced in these examples then the potential is correct everywhere including at and near the boundaries.

Although the one-orbital formula is exact for one-electron and most closed-shell two-electron systems, it must be carefully implemented numerically to avoid spurious results. As seen in Appendix A (Target densities with numerical noise), noisy densities can easily produce wildly inaccurate potentials when using Equation 7 and finite difference operators. Such noisy behavior is unphysical because densities arising from the Schrödinger equation are theoretically continuous even when generated by singular potentials.[3] The following examples show how to appropriately smooth target densities based on the noise level before inserting them into the one-orbital formula.

We first illustrate smoothing using the particle-in-a-box[3] ground state density

$$n_0^{exact}(x) = \frac{2}{L}\sin^2\left(\frac{\pi x}{L}\right) \tag{9}$$

corresponding to the potential

$$v(x) = \begin{cases} 0 & \text{if } 0 < x < L \\ \infty & \text{otherwise} \end{cases}. \tag{10}$$

If we take the exact ground-state density n_0^{exact} and add a small amount of weighted noise generated from the standard normal distribution $N(\mu, \sigma^2)$[9] according to the formula

$$n_0^{noise} = n_0^{exact}\left[1 + N\left(0, 1 \times 10^{-10}\right)\right], \tag{11}$$

then the potential resulting from the one-orbital formula will also be noisy according to Equation (A2). This noise can be removed by fitting a cubic spline to the target density weighted by the inverse of the approximate standard deviation $w = 1/(\sigma * n_0^{approx})$. We implement this smoothing procedure using the `UnivariateSpline` routine in the SciPy library and the results are shown in Figure 3 for a box of length 5 with varying numbers of grid points. This example shows that smoothing is essential for dense grids with errors but not as important on coarse grids since the derivative of the error does not dominate in the one-orbital formula as explained in Appendix A (Finite difference operators).

The target density in the previous example was well approximated by a cubic spline but other target densities may need additional modifications before smoothing is applied. In particular, exponentially decaying densities are not adequately described by cubic splines unless they are first logarithmically transformed. We show how this can be done using the ground-state of the harmonic oscillator with the same weighted noise given by Equation 11. In this case we take the logarithm of the noisy density, fit this new quantity to a cubic spline using the weights $w = 1/\log(1+\sigma)$, and then exponentiate the result to get the smoothed target density. The potential produced using this logarithmic scaling before fitting is much more accurate in the asymptotic region than the potential produced with no scaling before smoothing as seen in Figure 4. We refer to this smoothing procedure later in this tutorial as the logarithmic smoothing method. Even highly singular potentials can be recovered using this method provided that the locations of zero density are first averaged with their nearest neighbors before applying the scaling and smoothing. (This or a similar procedure must be followed to avoid division-by-zero errors.) Figure 4 shows the example described in Ref. [36] in which the density of the first excited state of the simple harmonic oscillator is treated as a ground state density. The singular feature of the potential $(\delta(x)/|x|)$ becomes more pronounced with this procedure as the grid is refined.

In the previous examples, we were able to correctly smooth the target densities by having estimates of the standard deviation at each grid point. Such information would be typical of an experimentally determined density but most of the error patterns encountered in our density-to-potential inversions come from computationally determined densities using interacting wave function methods. In these cases, we can use error estimates to compute the correct weights for the smoothing procedure. We show how this can be done by computing the ground state density of the harmonic oscillator using the infinite-to-finite spatial grid mapping of Ref. [27] with a scaling parameter of $\alpha = 2$. (We purposely use a different grid to compute the target density to avoid comitting an inverse crime.) We compute the target density on this grid once with a second-order
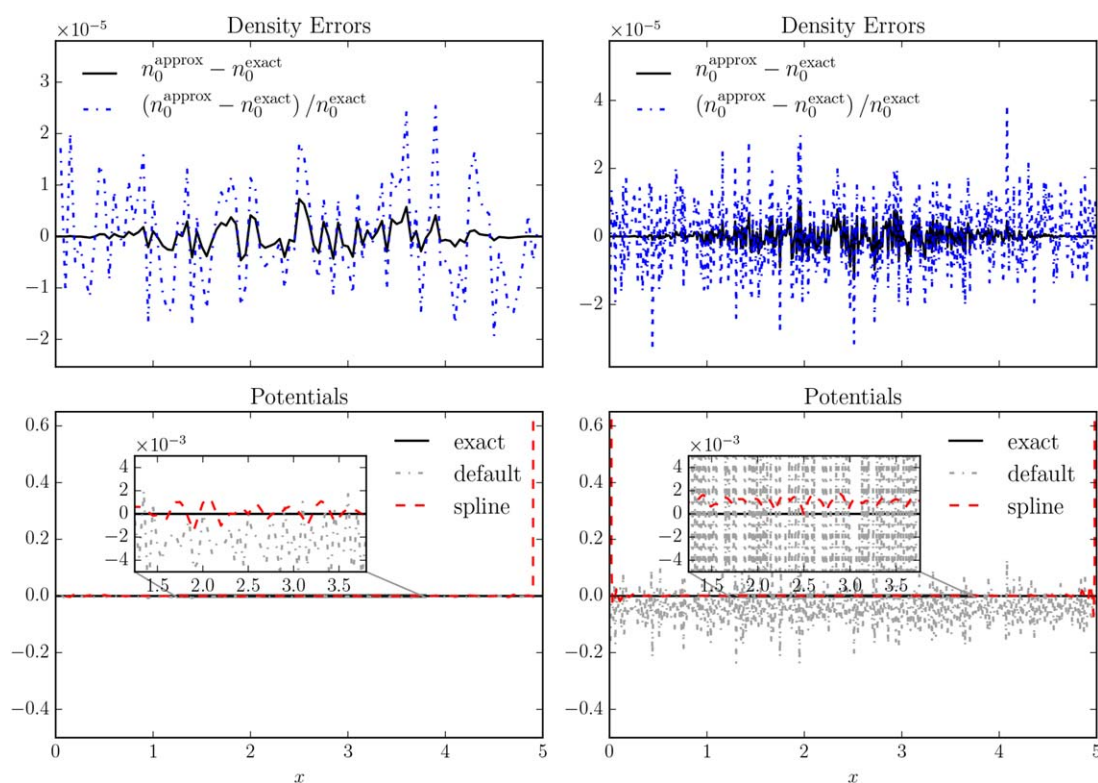
FIGURE 3 The ground-state density of a particle in a box is contaminated with weighted randomly distributed noise (top) and then inserted into the one-orbital formula before smoothing (default) and after smoothing (spline) via a weighted cubic spline (bottom). The smoothing has increasing importance as the number of grid points is increased from 101 (left) to 501 (right) even though the target density's error level is the same
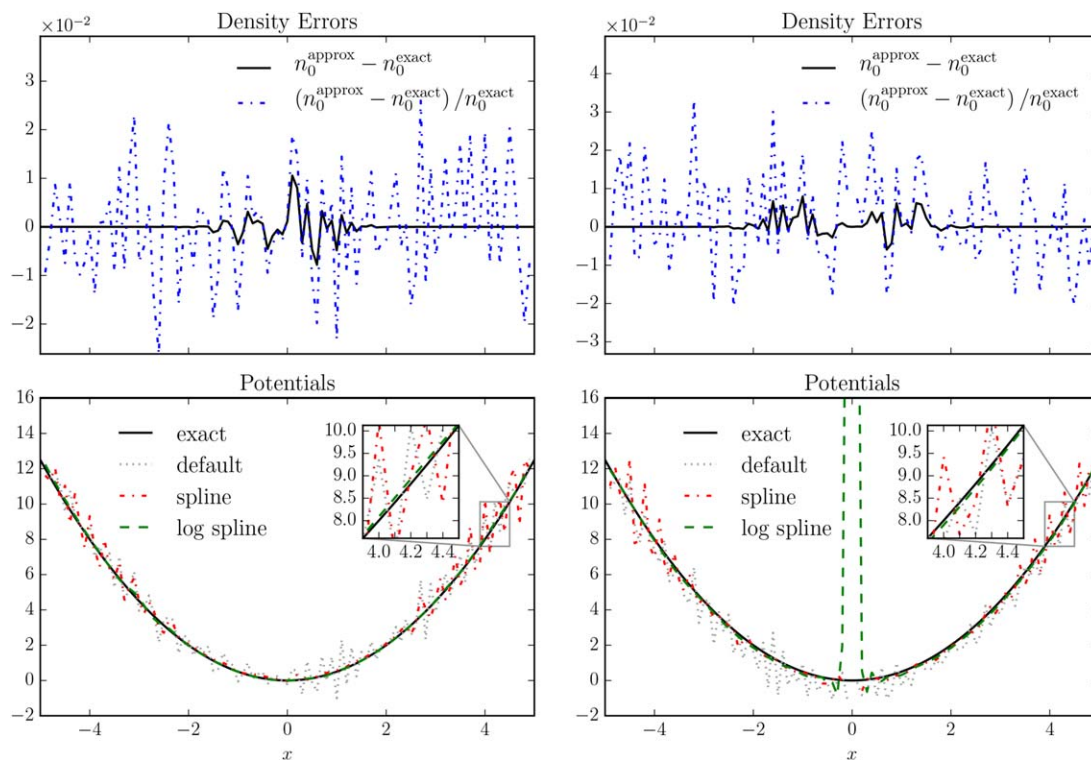

FIGURE 4 The ground-state density of the harmonic oscillator is contaminated with weighted randomly distributed noise (top left) and then inserted into the one-orbital formula without smoothing (default), with smoothing (spline), and with a logarithmic transform before smoothing (log spline) via a weighted cubic spline (bottom left). The process is repeated on the right for the density of the first excited state and the distinct delta-well singularity in the potential at $x = 0$ is recovered
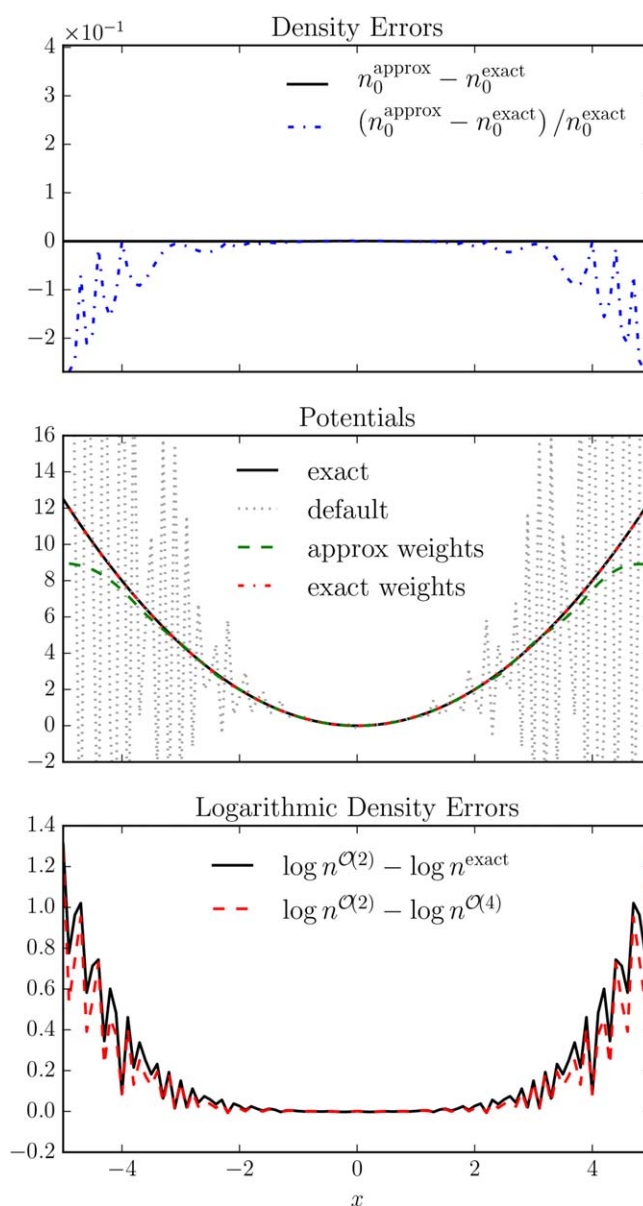
FIGURE 5 The ground-state density of the harmonic oscillator is computed using an infinite-to-finite spatial grid mapping (top) and then inserted into the one-orbital formula with both approximate and exact scaling weights (middle). The approximate weights are computed using an error estimate that tends to underestimate the error in the asymptotic region (bottom) and leads to an underestimated potential in that region

discretization of the Laplacian and 101 points, (labeled $n^{\mathcal{O}(2)}$), and then again with a fourth-order approximate Laplacian and 201 points, (labeled $n^{\mathcal{O}(4)}$). These two densities are linearly interpolated to the same equally spaced grid shown in Figure 4 and subtracted from each other to form an error estimate at each grid point. We then apply the logarithmically scaled spline fitting procedure described above with the approximate weights $w = 1/|\log n^{\mathcal{O}(2)} - \log n^{\mathcal{O}(4)}|$ and the exact weights $w = 1/|\log n^{\mathcal{O}(2)} - \log n^{\text{exact}}|$. The results of this calculation are displayed in Figure 5 and show that the error pattern of the approximate target density is far more systematic than the random noise error patterns in the previous examples. The smoothing using approximate weights underestimates the potential at the edges of the box because the error estimate is too small in that region. If the error estimate is computed using 501 points instead of 201 points in the fourth-order density computation then the potential is recovered correctly in the entire box. This example illustrates the need not only for error estimates but also for the need to put bounds on the error estimates either through computing additional error estimates or through statistical means.

Even if the target density is correctly smoothed before applying the one-orbital formula, the finite difference operator used to approximate the Laplacian can severely limit the accuracy of a density-to-potential inversion. As was the case with the spline fitting procedure described above for exponentially decaying densities, the finite difference method performs poorly in the asymptotic region unless extra precautions are taken. In subsection 2.2.3, we show how the finite difference operators can be modified to treat much of the exponential decay analytically through a scaling
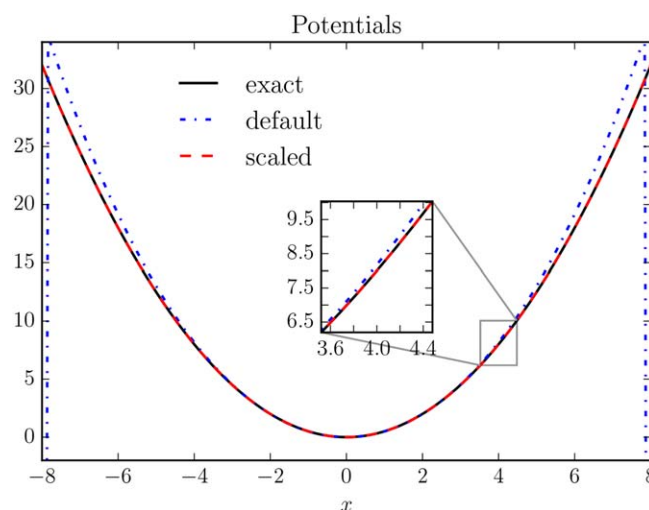
FIGURE 6 The ground-state density of the harmonic oscillator is inserted into the one-orbital formula without scaling (default) and with density scaling (scaled). The scaled finite difference operator performs much better in the asymptotic region $|x| > 4$ and does not have problems near the edges of the box

procedure. Figure 6 shows a comparison of the one-orbital formula applied to the exact ground state of the harmonic oscillator with and without scaling the finite difference operator.

Although the one-orbital formula is only exact for one- or two-electron systems, it is a remarkably good approximation for systems with large regions where the target density is dominated by one orbital. We illustrate the power of this approximation by applying it to a target density formed from the sum of the ground state and first two excited states of the harmonic oscillator. The harmonic potential is correctly recovered in the asymptotic region where most of the density comes from the second excited state as seen in Figure 7. The potential in the central region is not correct but it is a smooth approximation and is the same order of magnitude as the correct potential. Furthermore, the cost of this approximation is simply one sparse matrix-vector multiplication followed by an array division, which is negligible in comparison to the many sparse matrix-vector multiplications typically needed to solve the eigenvalue problem in Equation 4a.

The examples in this section demonstrate how the convergence criteria for DFT inverse problems are different from those of the direct problem. These examples and Appendix A clearly show that the size and location of the relative error in the target density is the main constraint on the accuracy of the potential resulting from a density-to-potential inversion. In many cases the target density must first be smoothed before performing an inversion and its standard deviation is the key ingredient in performing this smoothing operation correctly. We chose to use a spline-based smoothing filter because it is easy to incorporate the standard deviation but other methods can be used with similar success such as the Savitzky–Golay filter.[37] The accuracy of the finite difference operators also puts a strict limit on the quality of the recovered potential as some operators simply are not good approximations in certain regions of rapidly changing density.

2.2.2 | Standard methods

A number of density-to-potential inversion methods have been developed besides the one-orbital inversion formula. In this section we look at the iterative inversion method of Ref. [15] and the direct optimization method of Ref. [17]. A more complete review of the available density-to-potential inversion methods is found in Ref. [17]. The two inversion schemes described in this section are fairly representative of the available methods and serve as useful benchmarks later in this chapter. The target density in the remainder of this work is written as $\tilde{n}$ to distiguish it from the intermediate densities created in the iterative procedures that we describe here. When it does not lead to confusion, we also drop the subscript KS on the KS potential $v_{KS}(\mathbf{r})$ to simplify the presentation.

Van Leeuwen and Baerends: The iterative inversion method of van Leeuwen and Baerends described in Ref. [15] is derived by multiplying both sides of Equation 4a by ϕ_j^*, summing over j, and then dividing by the density given in Equation 4b to produce the formula

$$v(\mathbf{r}) = \frac{2}{n(\mathbf{r})} \sum_{j=1}^{N} \phi_j^*(\mathbf{r}) \frac{\Delta}{2} \phi_j(\mathbf{r}) + \varepsilon_j |\phi_j(\mathbf{r})|^2. \tag{12}$$

This formula is then turned into an iterative scheme by placing the target density in the denominator and iterating until convergence via the formula

$$v^{k+1}(\mathbf{r}) = \frac{2}{\tilde{n}(\mathbf{r})} \sum_{j}^{N} \frac{1}{2} \phi_j^{k*}(\mathbf{r}) \Delta \phi_j^k(\mathbf{r}) + \varepsilon_j^k |\phi_j^k(\mathbf{r})|^2 = \frac{1}{\tilde{n}(\mathbf{r})} \left[n^k(\mathbf{r}) v^k(\mathbf{r}) \right] = \frac{n^k(\mathbf{r})}{\tilde{n}(\mathbf{r})} v^k(\mathbf{r}), \tag{13}$$

where k is the iteration index. This iterative scheme simply increases $v^{k+1}(\mathbf{r})$ in regions where $n^k(\mathbf{r}) > \tilde{n}(\mathbf{r})$ and decreases it in regions where $n^k(\mathbf{r}) < \tilde{n}(\mathbf{r})$ as seen in the last line of Equation 13. The iterations are stopped when
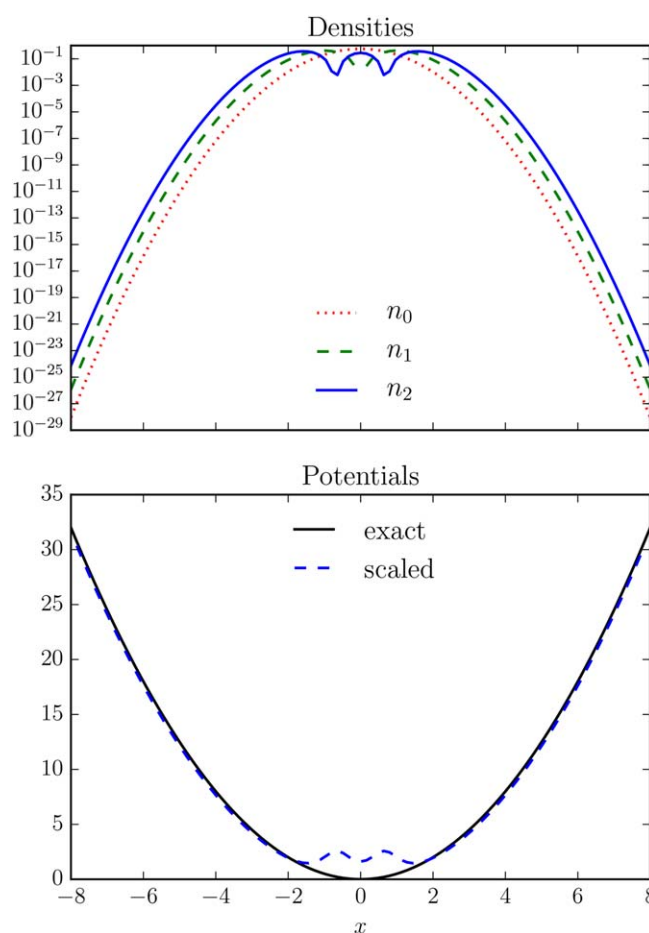
FIGURE 7 The densities of the ground-state and first two excited states of the harmonic oscillator (top) are summed and inserted into the one-orbital formula with density scaling (scaled). The resulting potential is an excellent approximation in the region $|x| > 2$ where the second excited state density is the dominant contribution to the total density

$$\max_{\mathbf{r}} \left| 1 - \frac{n^k(\mathbf{r})}{\tilde{n}(\mathbf{r})} \right| < \varepsilon \tag{14}$$

for some desired threshold ε. This scheme is sensitive to the initial potential guess $v^0(\mathbf{r})$ and usually requires a prefactor to avoid wild oscillations in the potential. Reference [15] suggests choosing this prefactor by enforcing the condition

$$1 - \delta < \gamma \frac{n^0(\mathbf{r})}{\tilde{n}(\mathbf{r})} < 1 + \delta, \tag{15}$$

where $\delta \approx 0.05$ and γ is the prefactor.

This iterative inversion scheme of van Leeuwen and Baerends is very easy to implement as it only requires a partial differential equation (PDE) solver to produce the new density $n^k(\mathbf{r})$ at each iteration. We do, however, make some modifications to the original scheme by not forcing the potential to approach any given limit at the boundaries and using the formula

$$v^{k+1}(\mathbf{r}) = v^k(\mathbf{r}) + \gamma \frac{n^k(\mathbf{r}) - \tilde{n}(\mathbf{r})}{\tilde{n}(\mathbf{r})} \tag{16}$$

instead of Equation 13. We do not enforce boundary conditions on the potential because many of the test cases we use recover the external potential with unknown boundary conditions as opposed to just the exchange-correlation potential with its known asymptotic behavior.[15] (Note that the orbitals may still have boundary conditions and the above discussion applies only to fixing values of the potential at the boundaries.) We use Equation 16 instead of Equation 13 because it still uses the same principle of increasing (decreasing) the potential in regions where the density is too large (small) but does not need boundary conditions to function properly. Other modifications are also possible such as the use of an approximate density–density response matrix to help guide the iterations.[38]

This method produces smooth potentials by construction provided that the underlying eigenvalue solver produces smooth orbitals. The reliance on a prefactor, however, requires some manual intervention or heuristics to ensure convergence. Large prefactors may result in wildly oscillating

potentials that never converge while small prefactors may produce very small potential changes that take an inordinate amount of time to converge. Choosing a prefactor is usually not a problem for isolated inversions but can become very tedious when repeatedly applying the procedure to cases with many similar inversions such as in the creation of dissociation curves.[39]

Wu–Yang Algorithm The Wu–Yang inversion algorithm first introduced in Ref. [17] is one of the more sophisticated and versatile DFT inversion methods. In this inversion algorithm the Levy constrained-search formulation of DFT is modified to produce an unconstrained optimization method. The functional

$$W_s[v(\mathbf{r})] = 2\sum_{j=1}^{N}\langle\phi_j|\hat{T}|\phi_j\rangle + \int d\mathbf{r}\, v(\mathbf{r})[n(\mathbf{r}) - \tilde{n}(\mathbf{r})] \tag{17}$$

is maximized to produce the optimal potential corresponding to the target density $\tilde{n}(\mathbf{r})$. (The kinetic energy is doubled because we are still assuming the orbitals are doubly occupied.) This maximization is usually performed using classical optimization routines and is very efficient because the required first derivatives (Jacobian values) are given by the simple formula

$$\frac{\delta W_s[v(\mathbf{r})]}{\delta v(\mathbf{r})} = n(\mathbf{r}) - \tilde{n}(\mathbf{r}). \tag{18}$$

There is very little extra computational effort required to compute these derivatives because the density is already computed at each iteration while computing Equation 17. The second derivatives (Hessian values) can also be computed either through first-order perturbation theory[17] or via the discrete adjoint-method described in Appendix B.

The Wu–Yang inversion algorithm was developed originally for the OEP method of DFT and later modified for the DFT inverse problem.[17,46] The method is efficient and robust when implemented properly but can suffer from rounding errors as shown at the end of Appendix A. It can also produce unphysical highly oscillatory potentials when an unbalanced basis set is used without regularization.[47]

2.2.3 | PDE-constrained optimization

The inversion schemes described in the previous section rely on modifications to the KS equations or underlying energy functionals. Such modifications are not essential in creating DFT inversion methods as we show in this section through the use of PDE-constrained optimization. Although conceptually simpler than many other inversion procedures, PDE-constrained optimization can be difficult to implement due to the large number of unknowns and programming challenges involved in the optimization procedure.[43] In this section, we give a brief introduction to PDE-constrained optimization in the context of DFT inverse problems and address many of these challenges. We also introduce an important scaling concept that plays a critical role in increasing the accuracy of the potential in the asymptotic regions of DFT inverse problems.

PDE-constrained optimization applied to the DFT inverse problem amounts to a fitting procedure in which the unknown potential is optimized until the corresponding density matches the target density. We use the weighted least-squares cost functional

$$F[v] = \frac{1}{2}||\sqrt{w(\mathbf{r})}[n([v],\mathbf{r}) - \tilde{n}(\mathbf{r})]||_2^2 \tag{19}$$

as our measure of fitness, where the subscript 2 indicates the L^2 norm and $w(\mathbf{r})$ is a positive definite weighting function.[48] The density $n([v],\mathbf{r})$ comes from solving Equations 4a and 4b for a given potential chosen during the optimization procedure. The numerical minimization of the discretized Equation 19 usually involves a very large number of unknown potential values and generally requires the use of gradient/Jacobian-based optimization algorithms.[49] The derivation, programming, and computation of these gradients, (functional derivatives of the cost functional given by Equation 19), is the main difficulty in applying PDE-constrained optimization to the DFT inverse problem.

We use the discrete adjoint method to compute the cost functional derivatives of Equation 19. This method is also called the discretize-then-differentiate method because the cost functional is first discretized and then differentiated with respect to the potential.[50] Appendix B contains a complete derivation of the discrete adjoint equations for the DFT inverse problem in multiple dimensions. These derivatives are then employed in the truncated-Newton (TN) algorithm of Ref. [10] as implemented in the SciPy library[32] to optimize the unknown potential until a desired density fit is found. The cost of computing the functional derivatives via the discrete adjoint-method is roughly the same as computing $F[v]$ itself depending on the choice of linear solver.

The PDE-constrained inversion method outlined above can also suffer from rounding errors similar to the Wu–Yang algorithm described in the previous section. These errors can be avoided simply by weighting the densities in the asymptotic region more heavily than in the high-density regions. One possibility is to set $w(\mathbf{r}) = 1/[\tilde{n}(\mathbf{r})]^2$ so that the relative error is minimized instead of the absolute error. This weighting scheme in combination with the rescaling described below allows us to accurately reproduce the unknown potential in all regions of a given density-to-potential problem.

As mentioned in subsection 2.2.1, the finite difference operators used to approximate the Laplacian are not very accurate in regions where the exponential decay dominates the behavior of the KS orbitals. The finite difference operators would be accurate in all regions if we could apply them to the logarithm of the orbitals similar to the logarithmic spline fitting procedure described earlier. The KS orbitals, however, are not positive semidefinite like the particle density so we instead write them as $\phi_j(\mathbf{r}) = \sqrt{\tilde{n}(\mathbf{r})}g_j(\mathbf{r})$ and solve for the scaled orbitals $\{g_j(\mathbf{r})\}$ in the KS equations
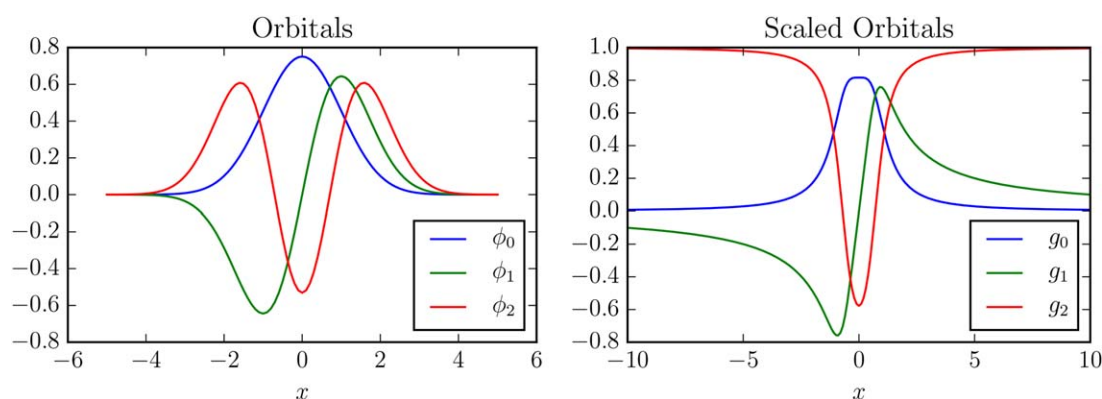
FIGURE 8 The first three orbitals of the one-dimensional simple harmonic oscillator without scaling (left) and with scaling by the square root of the molecular density (right)

$$\varepsilon_j \sqrt{\tilde{n}(\mathbf{r})} g_j(\mathbf{r}) = \left[-\frac{\Delta}{2} + v(\mathbf{r}) \right] \sqrt{\tilde{n}(\mathbf{r})} g_j(\mathbf{r}) \text{ and} \tag{20a}$$

$$n(\mathbf{r}) = \tilde{n}(\mathbf{r}) \sum_{j=1}^{N} |g_j(\mathbf{r})|^2. \tag{20b}$$

After distributing the Laplacian and canceling common factors, Equation 20a can be rewritten as

$$\varepsilon_j g_j(\mathbf{r}) = -\frac{1}{2} \left[\frac{\Delta \sqrt{\tilde{n}(\mathbf{r})}}{\sqrt{\tilde{n}(\mathbf{r})}} + 2 \frac{\nabla \sqrt{\tilde{n}(\mathbf{r})}}{\sqrt{\tilde{n}(\mathbf{r})}} \cdot \nabla + \Delta + v(\mathbf{r}) \right] g_j(\mathbf{r}). \tag{21}$$

The scaling factor can also be written as

$$\sqrt{\tilde{n}(\mathbf{r})} = \exp\left\{ \frac{1}{2} \log \tilde{n}(\mathbf{r}) \right\} \tag{22}$$

and differentiated to produce the terms

$$\frac{\nabla \sqrt{\tilde{n}(\mathbf{r})}}{\sqrt{\tilde{n}(\mathbf{r})}} = \frac{1}{2} \nabla \log \tilde{n}(\mathbf{r}) \text{ and} \tag{23}$$

$$\frac{\Delta \sqrt{\tilde{n}(\mathbf{r})}}{\sqrt{\tilde{n}(\mathbf{r})}} = \frac{1}{2} \left[\frac{1}{2} |\nabla \log \tilde{n}(\mathbf{r})|^2 + \Delta \log \tilde{n}(\mathbf{r}) \right]. \tag{24}$$

The derivatives of $\log \tilde{n}(\mathbf{r})$ and $\{g_j(\mathbf{r})\}$ are well approximated by the finite difference operators described in subsection 2.1 because they do not decay exponentially. An example of the scaled orbitals for a system consisting of the first three orbitals of the harmonic oscillator can be seen in Figure 8.

Although it is common to enforce boundary conditions on the orbitals when solving the KS equations, we simply let the numerical optimization routine choose the ideal boundary conditions via the DCBC method described in subsection 2.2.1. In Figure 8, the scaled orbitals appear to have derivatives of nearly zero near the boundaries so it may seem reasonable to apply zero derivative boundary conditions when using scaled orbitals. This assumption leads to errors near the boundaries as shown in Figure 9 in which the PDE-constrained optimization procedure outlined above is applied to a target density formed from the sum of the ground state and first two excited states of the harmonic oscillator. This example clearly shows that enforcing approximate zero derivative boundary conditions on the scaled orbitals results in the incorrect potential at the boundary whereas the potential is recovered correctly when no boundary conditions are applied. If the target density happens to be produced using artificial boundary conditions then the potential recovered when not applying boundary conditions on the orbitals may have several incorrect values at the edge of the simulation box.

2.2.4 | Constrained variational method

The previous inversion methods described in this chapter all require the solution of an eigenvalue problem at each iteration. In the case of small systems this is not a severe constraint but as the system size increases the solution of the eigenvalue problem begins to dominate the computation.[21] Although dismissed in Ref. [17] as "difficult to perform," the Levy constrained-search formulation can be used directly to perform a density-to-potential inversion without repeatedly solving eigenvalue problems at each iteration. In this section, we show how to overcome many of the difficulties involved in this constrained-search inversion method through the use of scaling and regularization.
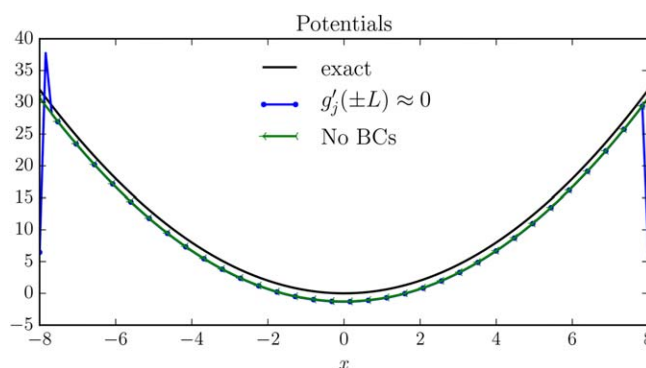
FIGURE 9 The potentials produced in a density-to-potential inversion using the inversion method described in subsection 2.2.3 and a target density formed from the sum of the ground state and first two excited states of the harmonic oscillator. The potential is correct at the boundaries when no boundary conditions are applied to the scaled orbitals (No BCs) and incorrect when the first derivative of the scaled orbitals is set to zero at the boundaries ($g'_j (\pm L) \approx 0$)

In the Levy constrained-search formulation of DFT, the noninteracting kinetic energy is minimized subject to the constraints that the orbitals are orthonormal and produce the molecular density of the real system. The functional to be minimized in this search can be written using Lagrange multipliers as

$$J(\{\phi_j\}) = \int d\mathbf{r} \left\{ \sum_{j=1}^{N} |\nabla\phi_j|^2 + v_{KS}(\mathbf{r}) \left[\sum_{j=1}^{N} \phi_j^2(\mathbf{r}) - \tilde{n}(\mathbf{r}) \right] + \sum_{j=1}^{N} \sum_{k=j}^{N} \varepsilon_{j,k} \left(\phi_j(\mathbf{r})\phi_k(\mathbf{r}) - \delta_{j,k} \right) \right\}, \tag{25}$$

where the orbitals are assumed to be real for simplicity of presentation. The first term in Equation 25 is one possible form of the kinetic energy and lacks a factor of 1/2 because the orbitals are doubly occupied. This expression can be used directly in a constrained numerical optimization program to find the orbitals and KS potential for a given target density without the need to solve an eigenvalue problem at each iteration. The orbitals themselves are being varied by the optimizer to minimize the kinetic energy while satisfying the density and orthonormality constraints. (This inverse problem method is similar to the gradient-based forward method described in Ref. [34].) It does, however, suffer from the same rounding errors involved in the Wu–Yang algorithm as it still involves a minimization of the noninteracting kinetic energy. Furthermore, the orthonormality and density constraints can easily lead to unphysical orbitals as they compete with one another in the optimization.

To resolve the issues aforementioned, we use the scaled orbitals introduced in subsection 2.2.3 and regularize them by adding the penalty functional

$$R(\{g_j\}, \alpha) = \alpha \sum_{j=1}^{N} \int d\mathbf{r} |\nabla g_j(\mathbf{r})|^2, \tag{26}$$

to the cost functional $J(\{\phi_j\})$, where $\alpha > 0$ is the regularization parameter chosen via the discrepancy principle.[12] The density constraint with the scaled orbitals is

$$\int d\mathbf{r} \left[\sum_{j=1}^{N} g_j^2(\mathbf{r}) \right] - 1. \tag{27}$$

No boundary conditions are applied to the orbitals for reasons explained in subsection 2.2.3.

Optimizing the noninteracting kinetic energy using the scaled and regularized orbitals produces a set of orthonormal orbitals that are a unitary transformation of the KS orbitals.[51] We use the Ipopt library[52] to perform this constrained minimization. The KS potential can be extracted from these orbitals by solving the set of linear equations

$$\left\{ \int d\mathbf{r} \, \phi_\beta(\mathbf{r})\Delta\phi_\alpha(\mathbf{r}) = 2 \int d\mathbf{r} \, \phi_\beta(\mathbf{r})\phi_\alpha(\mathbf{r})v_{KS}(\mathbf{r}) + \left(\sum_{m=1}^{\alpha} \varepsilon_{m,\alpha}\phi_m(\mathbf{r}) + \sum_{n=\alpha}^{N} \varepsilon_{\alpha,n}\phi_n(\mathbf{r}) \right) \right\}, \tag{28}$$

where $\alpha, \beta = 1, \ldots, N$. This linear system can be solved efficiently using a sparse least-squares solver in which the full matrix is never explicitly created. An example of this density-to-potential inversion procedure with two different regularization parameters is shown in Figure 10 for a target density formed from the sum of the ground state and first two excited state densities of the harmonic oscillator.

3 | INVERSION EXAMPLES

We have chosen to use well-known noninteracting quantum systems with which most readers are familiar to help illustrate the inversion methods described in subsection 2.2. The KS potentials recovered for these systems are equal to the known external potential because the systems are
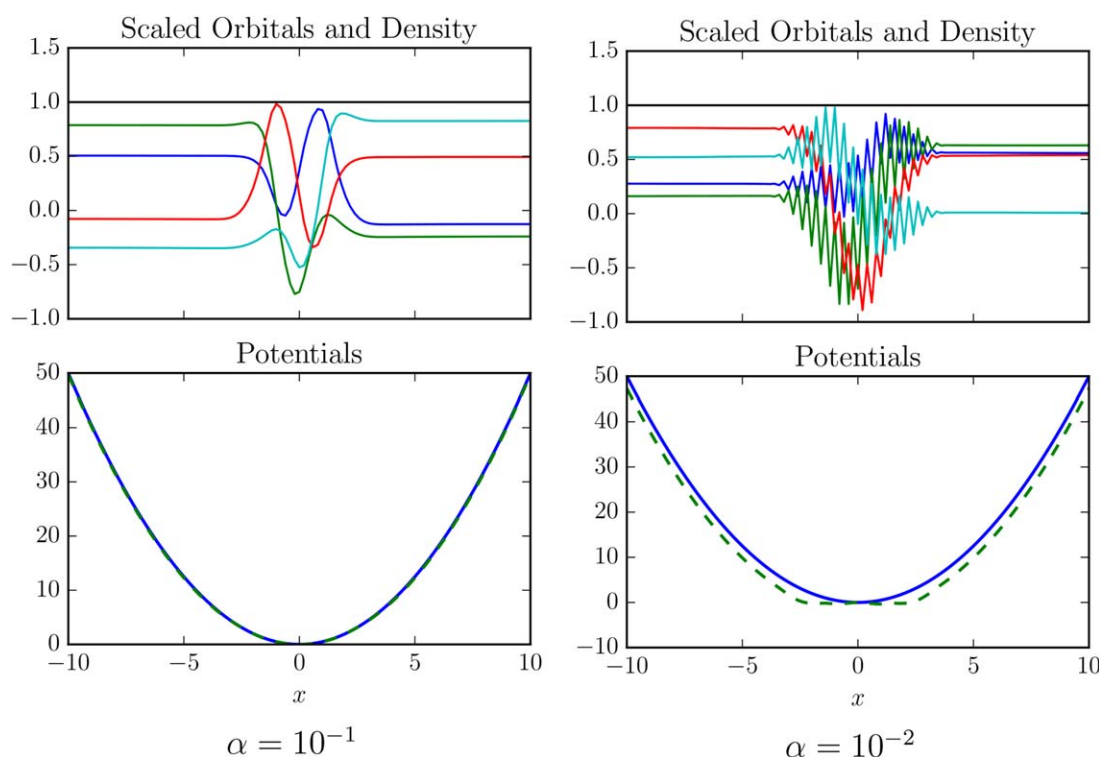
FIGURE 10   The scaled orbitals and density (top) of two constrained variational density-to-potential inversions with $\alpha=10^{-1}$ (left) and $\alpha=10^{-2}$ (right). The scaled density is the flat black line with a value of 1 for all x. The orthonormalization constraint dominates in the under-regularized example (right) producing wild oscillations in the high-density region. The corresponding potentials (bottom) are found by solving Equation 28

noninteracting and this allows us to focus on the numerical properties of each inversion algorithm. More realistic DFT inversions use interacting densities and extract the exchange-correlation potential from the recovered KS potential using Equation 4c. Unless otherwise noted, the examples we present all use the one-orbital approximation as the initial guess for the unknown potential. The gradients used for the PDE-constrained optimizations are given in Appendix B. Table 1 displays the abbreviations used in this section to distinguish the different inversion methods.

In these examples, we implement scaling, different finite difference approximations, and the DCBC method for both the PDE and CV routines. We do this simply because we are more familiar with these methods and already have this capability implemented in our code. It should also be possible to modify both the vLB and WY methods in similar ways to implement these features so that all of the schemes have similar error patterns in the asymptotic regions. All of the error patterns in these examples are shifted relative to one another to see the differences and make better comparisons.

3.1 | Harmonic potential inversion

Our first inversion example is a noninteracting system of six electrons in the harmonic potential $v(x)=\frac{1}{2}x^2$. The target density is constructed from the lowest 3 orbitals

$$\phi_0(x)=\pi^{-1/4}e^{-\frac{x^2}{2}}, \tag{29}$$

$$\phi_1(x)=\frac{\sqrt{2}x}{\pi^{1/4}}e^{-\frac{x^2}{2}}, \text{and} \tag{30}$$

$$\phi_2(x)=\frac{(2x^2-1)}{\sqrt{2}\pi^{1/4}}e^{-\frac{x^2}{2}}, \tag{31}$$

TABLE 1   The labels used to identify each DFT inversion method in section 3

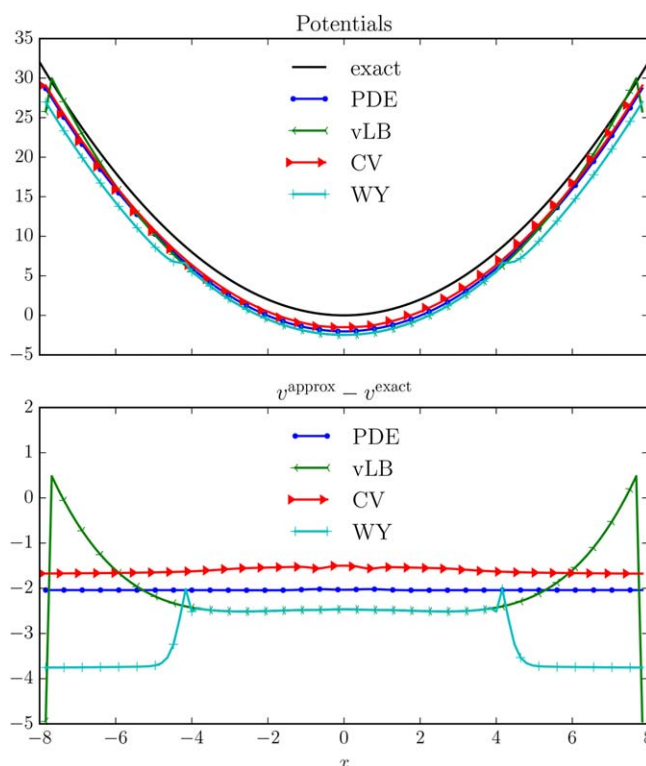| Inversion method	Label
PDE-constrained optimization	PDE
van Leeuwen and Baerends	vLB
Wu and Yang	WY
Constrained Variational	CV

FIGURE 11 The potentials produced in a density-to-potential inversion using a target density formed from the sum of the ground state and first two excited states of the harmonic oscillator (top). The correct potential is only unique up to a constant so we also plot the difference $v^{\text{approx}} - v^{\text{exact}}$ (bottom) with the more constant lines being an indication of correctness

with each orbital doubly occupied as indicated in Equation 1b. The potentials recovered from this density using several different inversion methods are shown in Figure 11 using a grid of 101 equally spaced points running from -8 to 8. The potential is only unique up to a constant so we also plot the differences between the recovered and exact potentials with the more constant differences indicating better recovered potentials.

In this example, the rounding errors in the WY algorithm's functional dominate in the region $|x| > 4$ as explained in Appendix A (Rounding Errors). The unscaled finite-difference scheme used in the vLB method is not accurate in the region $|x| > 4$ and leads to an incorrect higher curvature in that region. Both the CV and PDE methods have similar problems when they are not scaled or use zero boundary conditions so this example is not meant to show the superiority of any one scheme but rather to show the importance of these modifications.

3.2 | Morse potential inversion

Our second inversion example is a noninteracting system of six electrons in the Morse potential

$$v(r) = D_e \left(1 - e^{-a(r-r_e)} \right),$$ (32)

where $a = 1/2$, $D_e = 10$, and $r_e = 3$. Analytic solutions for this system's wave functions can be found in Ref. [53] and are used as input in Equation 1b for forming the target density. The potentials recovered from the target density described above are shown in Figure 12 using a grid of 51 equally spaced points running from 1 to 9.

This inversion example illustrates some of the numerical issues described in section 2. One numerical issue is the need to adequately represent the kinetic energy operator on the chosen grid. The PDE and CV methods in this example use a fourth-order approximation of the Laplacian while the other methods use a second-order approximation. This difference in accuracy on such a coarse grid leads to several wiggles in the WY and vLB potentials because the inversion routines fit to the errors in the approximate Laplacian. Additionally, the one-orbital approximation is not a sufficiently accurate initial guess for the scaled PDE-constrained optimization to locate the correct potential, hence the optimizer gets stuck in a local minimum. (The potential is so shallow that the scaled density is much too large near the right boundary.) This can be remedied by either raising the boundary values of the initial guess to force a more rapid decay or by including a small amount of regularization via Equation 26 with $\alpha = 1$ to guide the initial optimization and then turning it off with $\alpha = 0$ before running the final optimization. The PDE result shown in Figure 12 uses the regularization option but the result is nearly identical to that found when raising the boundary values of the initial guess. Both of these issues arise even when we have an analytic target density with no added errors.
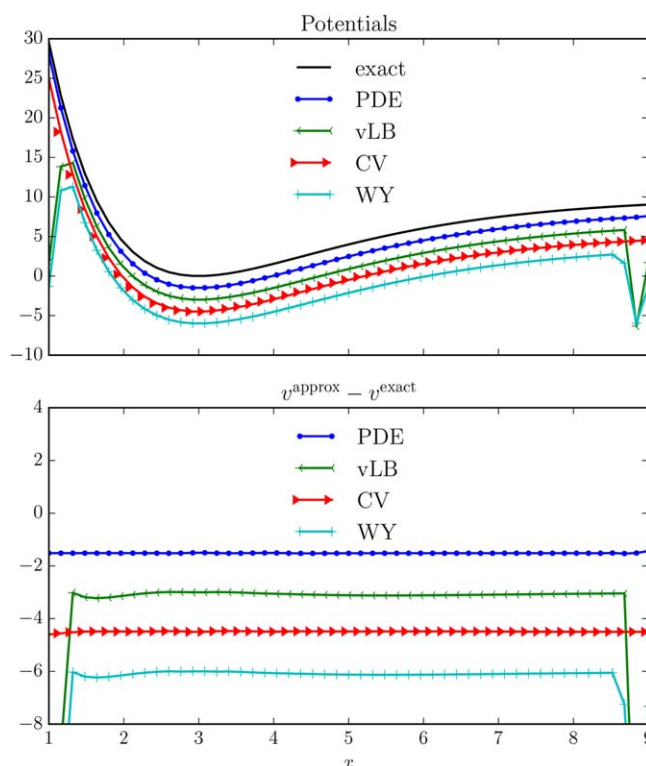
FIGURE 12 The potentials produced in a density-to-potential inversion using a target density formed from the sum of the ground state and first two excited states of the Morse oscillator (top). The correct potential is only unique up to a constant so we also plot the difference $v^{approx} - v^{exact}$ (bottom) with the more constant lines being an indication of correctness. Both the PDE and CV methods are scaled and use fourth-order approximate Laplacians compared to the second-order schemes used in the vLB and WY methods

3.3 | Kronig–Penney inversion

Our last inversion example is a noninteracting system of six electrons per unit cell in the Kronig–Penney potential

$$v(x) = \begin{cases} V_0 & \text{if } mL - \dfrac{a}{2} \leq x \leq mL + \dfrac{a}{2} \\ 0 & \text{otherwise} \end{cases}, \tag{33}$$

where $L = 2$ is the length of the unit cell, $a = 5/4$ is the width of the well, $V_0 = -10$ is the depth of the well, and m is an integer. The target density is found by extracting the density from the center cell of a finite system comprised of 15 unit cells, ($m = -7, \ldots, 7$ in Equation 33), with 400 points per unit cell and a second-order discretization of the Laplacian. By performing a finite calculation of the target density we avoid the risk of committing an inverse crime.

The inverse problem for this example is solved on a grid with 50 points using PDE-constrained optimization without scaling and a fourth-order approximate Laplacian. (We do not use scaling because the exponential decay does not dominate anywhere in the system's density.) The Hamiltonian is modified as explained in Ref. [22] to work with the Bloch functions $\{\phi_{i,\mathbf{k}}(\mathbf{r})\}$. In particular, the kinetic energy operator has a k-space dependence

$$\hat{T}_{\mathbf{k}} = -\frac{1}{2}\left(\Delta + 2\imath\mathbf{k}\cdot\nabla - k^2\right) \tag{34}$$

and the total density is found via an integration over the first Brillouin zone

$$n(\mathbf{r}) = \frac{1}{\Omega_{\mathbf{k}}}\int_{\Omega_{\mathbf{k}}} d\mathbf{k}\sum_{i=1}^{N_{orbs}} |\phi_{i,\mathbf{k}}(\mathbf{r})|^2, \tag{35}$$

where $\Omega_{\mathbf{k}}$ is the volume of the first Brillouin zone. We perform this k-space integration using the trapezoid rule and 11 equally spaced points running from $-\pi/L$ to π/L. (For simplicity we integrate over the whole first Brillouin zone but usually symmetry is used to reduce this to an integral over the irreducible Brillouin zone.) The results of this inversion are shown in Figure 13 and show that even sharp features such as Heaviside functions can be recovered successfully in these inversions.
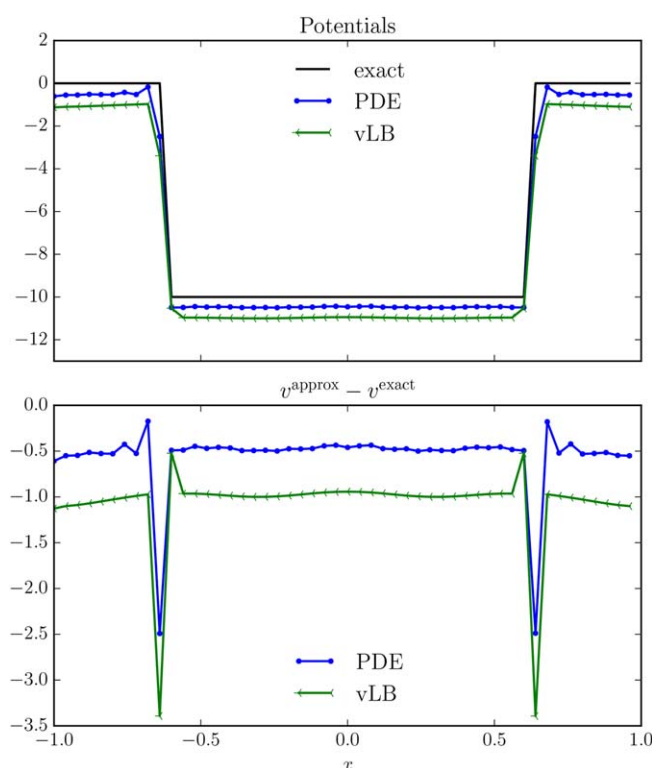
FIGURE 13 The potentials produced in a density-to-potential inversion using a target density formed from the sum of the ground state and first two excited states of the Kronig–Penney periodic system (top). The difference $v^{approx} - v^{exact}$ (bottom) shows that most of the error is near the jumps at $-a/2$ and $a/2$

4 | CONCLUSIONS

We have presented a series of algorithms for performing density-to-potential inversions in DFT. Although the examples in this work are all one-dimensional model systems, the algorithms are written in multiple dimensions and are applicable to more realistic systems. In particular, these methods can all be used to find the exact exchange-correlation potential corresponding to a given interacting density as explained in section 2. We have also presented a detailed numerical analysis that points to the relative error as the main limitation in density-to-potential inversions and the need for error estimates in performing these inversions properly.

The inversion methods presented in this tutorial use the finite difference method for solving the governing PDEs but they can also be applied to the common basis function methods described in Ref. [1]. The insight we have gained through our many test cases and numerical analysis with the finite difference method can also be used to improve other inversion methods as hinted at in subsection 3.1. In particular, the scaled-orbital regularization presented in subsection 2.2.4 is perhaps a more natural regularization than the potential regularization presented in Ref. [47]. This regularization should also make it possible to examine more realistic examples including possibly ill-posed densities from experimental data.

All of the inversion methods used in section 3 have strengths and weaknesses that should be considered when choosing which method to use for a given inverse DFT problem. The PDE-constrained optimization algorithm is very much a black-box method requiring almost no physical intuition but it is only competetive with other methods when the gradient is computed efficiently and used in a good optimization routine. This is not a trivial endeavor as both the derivation and programming of these gradients can be quite involved as shown in Appendix B. Despite this weakness, it is very straightforward to adapt this method to different inverse problems resulting from extensions of DFT such as TDDFT.[43] The method of van Leeuwen and Baerends does have a sound physical motivation and is simple to program because it only needs to solve the Kohn–Sham equations. The biggest disadvantage to this method is its reliance on a prefactor, which requires user intervention and makes automating the routine difficult. The method of Wu and Yang is similar to the van Leeuwen and Baerends method in that it is also based on some physical motivation and only needs to solve the Kohn–Sham equations provided the Hessian is not needed. This method is very efficient when paired with a good optimizer but can suffer from rounding and overfitting errors if extra precautions are not taken. The constrained variational method is the only method that does not solve an eigenvalue problem at each iteration and this may be an advantage for large systems with more demanding eigenvalue problems. In the systems that we have studied this method appears to be insensitive to the initial guess with even random numbers for the initial orbitals resulting in rapid convergence. Its largest drawback is its current reliance on a regularization parameter but this choice can be automated using standard inverse

problem methods. Since we have only considered model one-dimensional systems in this study we do not know how these methods will perform on more realistic three-dimensional systems and plan to do future studies to see how they scale with larger systems.

Our original intent in developing inversion algorithms was to benchmark approximate exchange-correlation approximations to guide new approximations within DFT and Partition-DFT.[19] During this development, however, several new and interesting applications have come to light that warrant further study. For example, by allowing the inversion methods to enforce the boundary conditions by modifying the potential near the boundaries in the DCBC method, it should be possible to extract new approximate boundary conditions and compare them to other state-of-the-art methods.[28] Likewise, the orbital scaling introduced in subsection 2.2.3 could be modified for the direct DFT problem as an alternative to using adaptive grid methods.[22] Thanks to the very similar structure between the OEP method and the DFT inverse problem, algorithm development in one area can lead directly to improvements in the other as seen in the development of the Wu–Yang inversion algorithm from a previous OEP optimization method.[17] We plan to continue exploring these other areas of DFT and refine the density-to-potential inversion methods described in this tutorial.

ACKNOWLEDGMENTS

REFERENCES

[1] R. M. Martin, *Electronic Structure: Basic Theory and Practical Methods*, Cambridge University Press, Cambridge, UK; New York **2004**.

[2] K. Burke, L. O. Wagner, *Int. J. Quantum Chem.* **2013**, *113*, 96.

[3] E. Merzbacher, *Quantum Mechanics, 3rd ed.*, Wiley, New York **1998**.

[4] A. Kirsch, *An Introduction to the Mathematical Theory of Inverse Problems*, 2nd ed., Springer, New York **2011**.

[5] J. Hadamard, *Princeton Univ. Bull.* **1902**, *13*, 49.

[6] J. T. Chayes, L. Chayes, M. B. Ruskai, *J. Stat. Phys.* **1985**, *38*, 497.

[7] J. L. Müller, S. Siltanen, *Linear and Nonlinear Inverse Problems with Practical Applications*, Society for Industrial and Applied Mathematics, Philadelphia **2012**.

[8] T. E. Baker, E. M. Stoudenmire, L. O. Wagner, K. Burke, S. R. White, *Phys. Rev. B* **2015**, *91*, 235141.

[9] L. J. Bain, M. Engelhardt, *Introduction to Probability and Mathematical Statistics*, 2nd ed., Cengage Learning, Boston **2000**.

[10] S. Nash, *SIAM J. Numer. Anal.* **1984**, *21*, 770.

[11] M. J. D. Powell, *Comput. J.* **1964**, *7*, 155.

[12] C. R. Vogel, *Computational Methods for Inverse Problems*, Society for Industrial and Applied Mathematics, Philadelphia **2002**.

[13] R. C. Aster, C. H. Thurber, *Parameter Estimation and Inverse Problems*, Academic Press, Waltham, MA **2012**.

[14] Q. Zhao, R. C. Morrison, R. G. Parr, *Phys. Rev. A* **1994**, *50*, 2138.

[15] R. van Leeuwen, E. J. Baerends, *Phys. Rev. A* **1994**, *49*, 2421.

[16] L. O. Wagner, E. M. Stoudenmire, K. Burke, S. R. White, *Phys. Chem. Chem. Phys.* **2012**, *14*, 8581.

[17] Q. Wu, W. Yang, *J. Chem. Phys.* **2003**, *118*, 2498.

[18] M. Banafsheh, T. A. Wesolowski, *Int. J. Quantum Chem.* **2017**.

[19] J. Nafziger, A. Wasserman, *J. Phys. Chem. A* **2014**, *118*, 7623.

[20] S. Kümmel, J. P. Perdew, *Phys. Rev. B* **2003**, *68*, 035103.

[21] C. Le Bris, *Acta Numer.* **2005**, *14*, 363.

[22] A. Castro, H. Appel, M. Oliveira, C. A. Rozzi, X. Andrade, F. Lorenzen, M. A L. Marques, E. K. U. Gross, A. Rubio, *Phys. Status Solidi B* **2006**, *243*, 2465.

[23] X. Andrade, D. Strubbe, U. D. Giovannini, A. H. Larsen, M. J. T. Oliveira, J. Alberdi-Rodriguez, A. Varas, I. Theophilou, N. Helbig, M. J. Verstraete, L. Stella, F. Nogueira, A. Aspuru-Guzik, A. Castro, M. A. L. Marques, A. Rubio, *Phys. Chem. Chem. Phys.* **2015**, *17*, 31371.

[24] T. L. Beck, *Rev. Mod. Phys.* **2000**, *72*, 1041.

[25] B. Fornberg, *Math. Comput.* **1988**, *51*, 699.

[26] S. E. B. Nielsen, M. Ruggenthaler, R. van Leeuwen, [quant-ph] (**2014**), arXiv: 1412.3794, http://arxiv.org/abs/1412.3794

[27] F. Ladouceur, *Opt. Lett.* **1996**, *21*, 4.

[28] X. Antoine, A. Arnold, C. Besse, M. Ehrhardt, A. Schädle, *Commun. Comput. Phys.* **2008**, *4*, 729.

[29] P. Maragakis, J. Soler, E. Kaxiras, *Phys. Rev. B* **2001**, *64*, 193101.

[30] P. García-Risueño, J. Alberdi-Rodriguez, M. J. T. Oliveira, X. Andrade, M. Pippig, J. Muguerza, A. Arruabarrena, A. Rubio, *J. Comput. Chem.* **2014**, *35*, 427.

[31] M. A. L. Marques, M. J. T. Oliveira, T. Burnus, *Comput. Phys. Commun.* **2012**, *183*, 2272.

[32] E. Jones, T. Oliphant, P. Peterson, et al., *SciPy: Open Source Scientific Tools for Python (Version 0.16.1).* https://www.scipy.org/citing.html

[33] P. Pulay, *J. Comput. Chem.* **1982**, *3*, 556.

[34] X. Zhang, J. Zhu, Z. Wen, A. Zhou, *SIAM J. Sci. Comput.* **2014**, *36*, C265.

[35] I. G. Ryabinkin, V. N. Staroverov, *J. Chem. Phys.* **2012**, *137*, 164113.

[36] N. T. Maitra, K. Burke, *Phys. Rev. A* **2001**, *63*, 042501.

[37] A. Savitzky, M. J. E. Golay, *Anal. Chem.* **1964**, *36*, 1627.

[38] L. O. Wagner, T. E. Baker, E. M. Stoudenmire, K. Burke, S. R. White, *Phys. Rev. B* **2014**, *90*, 045109.

[39] J. Nafziger, A. Wasserman, *J. Chem. Phys.* **2015**, *143*, 234105.

[40] J. J. Alonso, J. E. Hicken, *J. Comput. Phys.* **2014**, *263*, 136.

[41] C. S. Rudisill, Y.-Y. Chu, *AIAA J.* **1975**, *13*, 834.

[42] N. P. van der Aa, H. G. ter Morsche, R. R. M. Mattheij, *Electron. J. Linear Algebra* **2007**, *16*.

[43] D. S. Jensen, A. Wasserman, *Phys. Chem. Chem. Phys.* **2016**, *18*, 21079.

[44] A. Castro, J. Werschnik, E. K. U. Gross, *Phys. Rev. Lett.* **2012**, *109*, 153603.

[45] M. B. Giles, N. A. Pierce, *Flow Turbulence Combustion* **2000**, *65*, 393.

[46] W. Yang, Q. Wu, *Phys. Rev. Lett.* **2002**, *89*, 143002.

[47] F. A. Bulat, T. Heaton-Burgess, A. J. Cohen, W. Yang, *J. Chem. Phys.* **2007**, *127*, 174101.

[48] C. R. Vogel, J. G. Wade, *SIAM J. Control Optim.* **1995**, *33*, 227.

[49] L. T. Biegler, O. Ghattas, M. Heinkenschloss, B. V B. Waanders, in *Large-Scale PDE-Constrained Optimization* (Eds: L. T. Biegler, M. Heinkenschloss, O. Ghattas, B. v. B. Waanders), no. 30 in Lecture Notes in Computational Science and Engineering, Springer Berlin Heidelberg **2003**, pp. 3–13, ISBN 978-3-540-05045-2 978-3-642-55508-4, http://link.springer.com/chapter/10.1007/978-3-642-55508-4_1

[50] J. E. Hicken, D. W. Zingg, *J. Comput. Phys.* **2014**, *256*, 161.

[51] R. G. Parr, W. Yang, *Density-Functional Theory of Atoms and Molecules*, Oxford University Press, USA **1994**.

[52] A. Wächter, L. T. Biegler, *Math. Program.* **2005**, *106*, 25.

[53] J. P. Dahl, M. Springborg, *J. Chem. Phys.* **1988**, *88*, 4535.

AUTHOR BIOGRAPHIES

DANIEL JENSEN is a postdoctoral appointee at Sandia National Laboratories. He received his B.Sc. and M.Sc. degrees from Brigham Young University. He obtained his Ph.D. degree from Purdue University under the guidance of Adam Wasserman.

ADAM WASSERMAN is an Associate Professor of Chemistry at Purdue University. He obtained his B.Sc. degree from the National University of Colombia in Bogotá, and his Ph.D. degree from Rutgers University under the guidance of Kieron Burke. Before joining the faculty at Purdue, he was a postdoctoral fellow with Eric Heller at Harvard University. Adam is a 2012 NSF-CAREER recipient, 2013 Alfred P. Sloan Research Fellow, and 2014 Camille Dreyfus Teacher-Scholar.

APPENDIX A

NUMERICAL ISSUES

A number of numerical problems arise in performing DFT inversions that are often very difficult to isolate. In this appendix, we use very basic one-dimensional models to show how boundary conditions, relative error, and rounding errors all set convergence limits on density-to-potential inversions. We employ the one-orbital formula shown in Equation 7 and a grid ranging from -4 to 4 with 51 points in the examples of this section unless otherwise noted.

Boundary conditions and inverse crimes

Many of the inversion schemes mentioned in subsection 2.2 impose boundary conditions on the orbitals that do not have to agree with the target density's boundary conditions. A mismatch in boundary conditions can place severe constraints on the accuracy of density-to-potential inversions in the boundary regions as we illustrate here with an example. For simplicity, we take the exact density of the simple harmonic oscillator's ground state
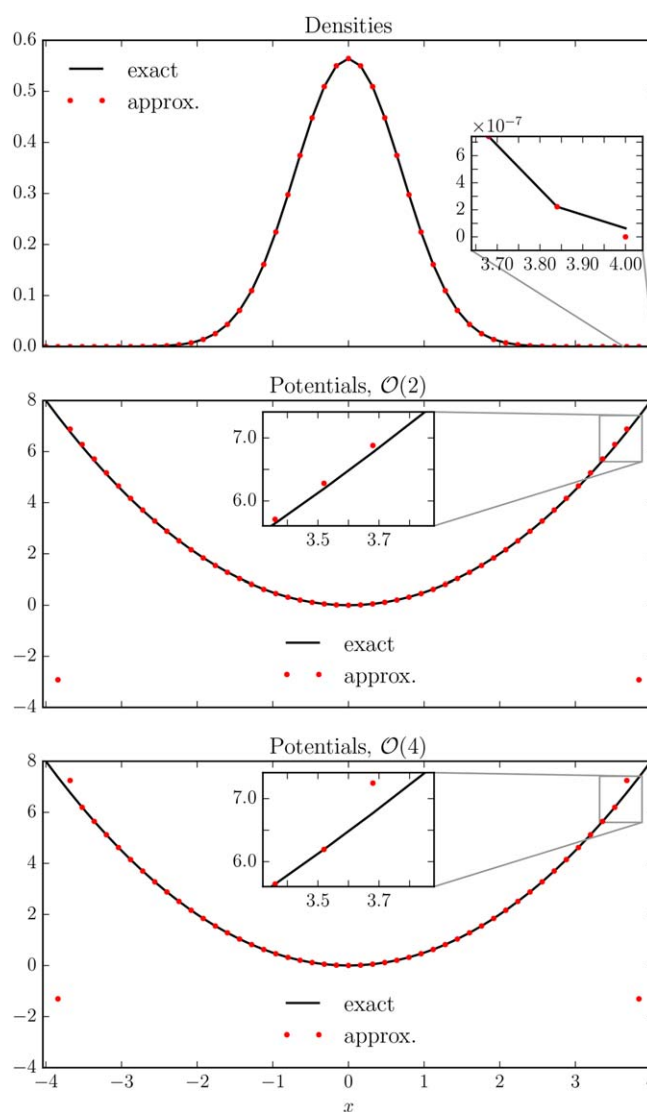

FIGURE A1 The degradation of a density-to-potential inversion when forcing incorrect boundary conditions on the orbitals. The exact ground-state density of the simple harmonic oscillator is modified to be exactly zero at the boundaries as commonly occurs when box-type (Dirichlet) boundary conditions are enforced (top). The potentials produced from this density with the one-orbital inversion formula and a second-order discretization of the Laplacian is incorrect near the boundaries (middle). (The outermost points have the largest error and we zoom in on the next closest values to show their errors.) The errors are even larger when a fourth-order approximation to the Laplacian is used (bottom)
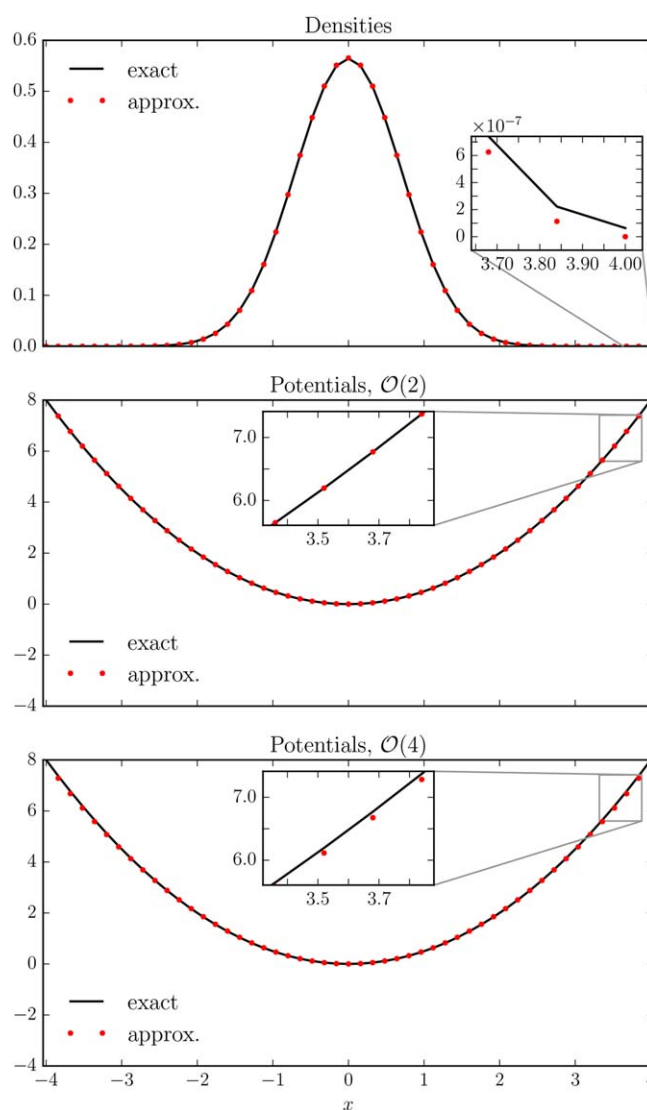
FIGURE A2 The frailty of inverse crimes in a density-to-potential inversion. The ground state density of the harmonic oscillator is computed numerically using a second-order approximation to the Laplacian and zero boundary conditions (top, approx). Applying the one-orbital formula to this density with a second-order approximate Laplacian produces nearly the exact potential (middle) and is an inverse crime. A fourth-order approximation in the one-orbital formula actually gives worse values most noticeable near the boundaries (bottom)

and set the values at the boundaries to zero as would be typical for box-type (Dirichlet) boundary conditions. The one-orbital formula should give us the exact potential for this problem everywhere except the boundaries, (where the density was artificially set to zero), but Figure A1 shows that the potential is also incorrect at several points next to the boundaries. The number of incorrect points near the boundaries increases as higher-order finite difference schemes are used to represent the Laplacian due to the increased nonlocality of the finite difference operators.

One common method for avoiding the mismatch in boundary conditions described above is to compute the target density with the same boundary conditions and numerical method. In Figure A2, we compute the ground state density of the harmonic oscillator numerically with box-type boundary conditions and a second-order finite difference approximation for the Laplacian. If the same finite difference approximation is used in the one-orbital formula then the resulting potential is almost exact everywhere. This type of inversion is known as an inverse crime. It is easily seen in Figure A2 that changing to a fourth-order approximate Laplacian with the same box-type boundary conditions does not have the same error cancellation and produces worse values than the second-order method.

The term "inverse crime" is commonly used in inverse problem theory to describe an inversion method relying on a cancellation of errors between the data simulation and reconstruction methods.[7] Inverse crimes are only committed in DFT inversions when the same numerical methods are used to solve both the direct and inverse problems. This is somewhat rare in performing DFT inversions because the densities being inverted are usually found using wave-function methods that employ different numerical procedures than those used in DFT inversions.[8] Inversion methods that rely on inverse crimes are very limiting in the number of systems that can be studied and will likely fail when applied to experimental data.
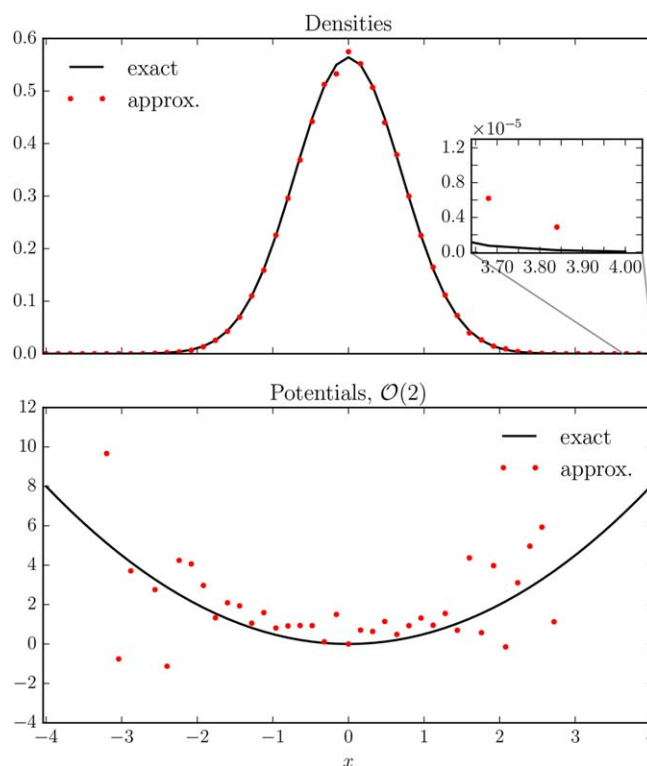
FIGURE A3 The effects of random noise in a density-to-potential inversion. When the ground state of the simple harmonic oscillator contains noise, (i.e., $\tilde{n}_0 = \left[\phi_0 + N\left(\mu=0, \sigma^2=2.5\times10^{-5}\right)\right]^2$), the resulting potential is very noisy everywhere but especially in the asymptotic region. The gauge in this example is fixed by forcing the value of the inverted potential to agree at $x = 0$ with the exact potential
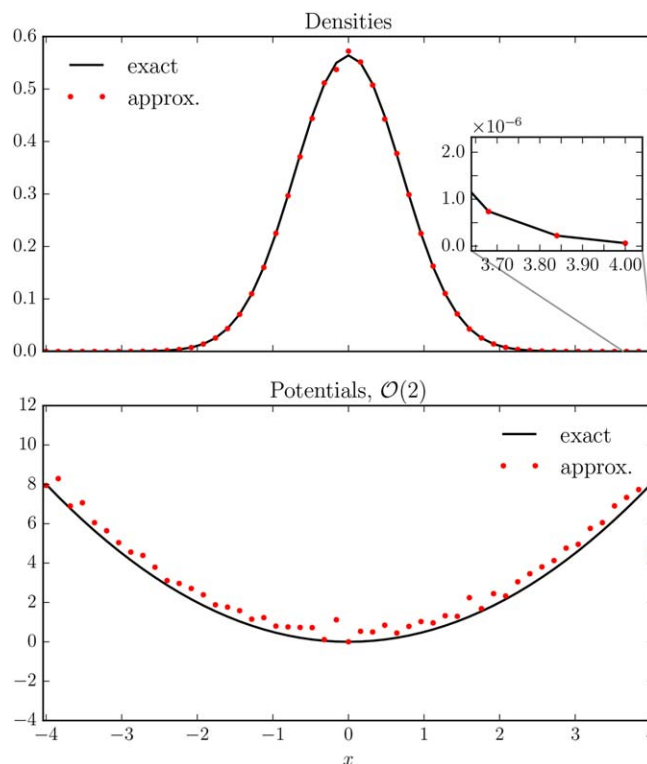

FIGURE A4 The effects of weighted random noise in a density-to-potential inversion. When the ground-state of the simple harmonic oscillator contains noise weighted by the orbital, (i.e., $\tilde{n}_0 = \{\phi_0\left[1 + N\left(\mu=0, \sigma^2=2.5\times10^{-5}\right)\right]\}^2$), the resulting potential is uniformly noisy. This indicates that the relative error in the target density sets a convergence limit in density-to-potential inversions. The gauge in this example is fixed by forcing the value of the inverted potential to agree at $x = 0$ with the exact potential
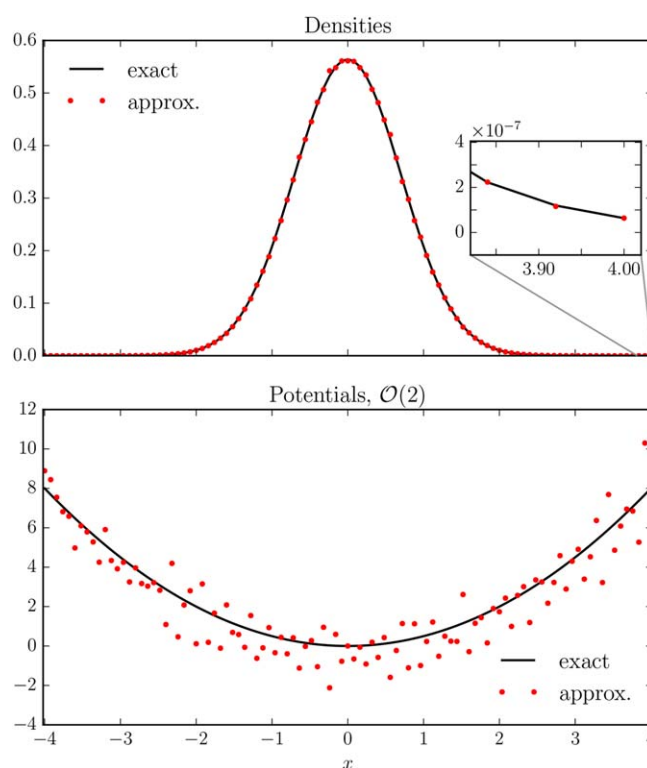
FIGURE A5 The effects of weighted random noise and increased grid resolution in a density-to-potential inversion. When twice the number of grid points are used for the example displayed in Figure A4 the resulting potential is actually worse and shows that increased resolution does not necessarily improve density-to-potential inversions. The gauge in this example is fixed by forcing the value of the inverted potential to agree at $x = 0$ with the exact potential

This example of an inverse crime is not meant to discourage practitioners from using knowledge about the target density in solving inverse DFT problems. Much of subsection 2.2.1 is devoted to using this knowledge to better approximate the true density and invert it to find the KS potential. An alternative is to use similar boundary conditions and numerical methods for both the interacting system and inverse problem with the understanding that there will be some cancellation of errors if the methods are similar enough. This is not really an inverse crime since the solution method for solving the interacting system is not identical to the numerical methods for solving the KS equations but it is not a requirement when solving these problems. In fact, the methods presented in subsection 2.2 can be applied to target densities resulting from a variety of different numerical methods with no need to rely on error cancellations.

Target densities with numerical noise

The examples in the preceding section assume a perfect knowledge of the target density but this is not a realistic assumption for most inverse problems. As mentioned in the previous example, the wave function numerical methods used in computing interacting densities are often very different from DFT numerical methods and the correct boundary conditions are not always known beforehand or may even be incorrect. It is far more realistic to assume that the target density in a density-to-potential inversion has some numerical error that must be accounted for to avoid overfitting. To illustrate the effects of numerical noise in the target density, we take the ground state orbital of the simple harmonic oscillator, add in a small amount of noise generated from the standard normal distribution $N(\mu, \sigma^2)$,[9] and put the density corresponding to this modified orbital in the one-orbital formula. The results of this inversion in Figure A3 are very poor especially in the asymptotic region. If we add a relative error to the ground state orbital instead of the uniform error described above then the resulting potential is still noisy but the noise level is now the same everywhere as seen in Figure A4. These two examples show that the relative error of the target density sets a limit on the convergence in density-to-potential inversions. Figure A5 shows that simply increasing the number of grid points actually makes the inverted potential worse when the same amount of error is present in the target density.

Finite difference operators

As mentioned in the previous examples, higher-order finite difference operators do not always improve density-to-potential inversions and boundary conditions often have a dramatic impact on the quality of an inversion. Another way to study the delicate balance between finite difference
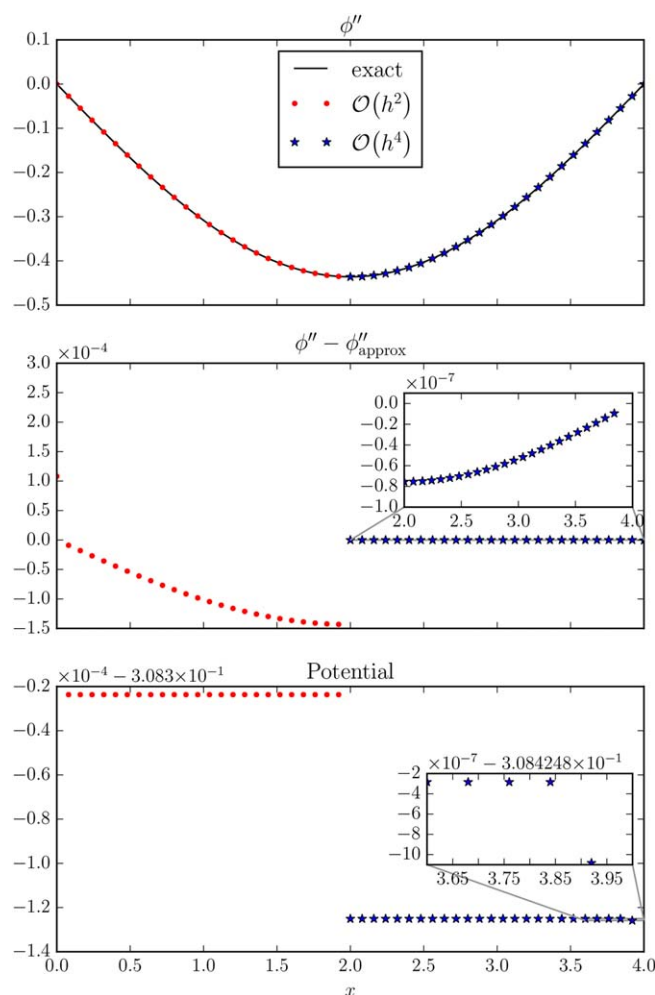
FIGURE A6 The effects of combining two different finite difference operators in a density-to-potential inversion. The second derivative is computed with a second-order method on the left side of the box and a fourth-order method on the right (top). The error patterns on both sides of the box are smooth but there is a distinct gap between them (middle). The resulting potential (bottom) also has a gap corresponding to the difference in error patterns and a small gap at the right boundary where a backwards finite-difference operator is used (bottom inset). The scale on the y-axis of the lower two plots is actually very small (10^{-4}) so these errors are not usually seen but they do make a difference in the inverse problem

operators and boundary conditions is to use the ground state density of a particle in a box in the one-orbital formula. If we compute the derivative in Equation 7 with one finite difference operator on the left side of the simulation box and a different operator on the right then the resulting second derivative has different error patterns on both sides of the box. The resulting potential has a jump between the left and right sides of the box as seen in Figure A6. This jump is actually very small, (on the order of 10^{-4} as seen in the scale), but makes a significant impact on the error pattern. Although one usually does not mix finite difference operators inside the simulation box, different operators are often used at the boundaries and produce similar jumps even if the operators are the same order of accuracy. An example of this jump at the boundary can be seen in Figure A6 at the right boundary where a fourth-order backwards finite-difference operator produces a different error pattern than the fourth-order centered operator used elsewhere. (See Equation 5 for the matrix representation of this fourth-order finite-difference operator.)

Many of the somewhat counterintuitive behaviors shown in the preceding examples can be understood mathematically by studying the effects of errors in the one-orbital formula. If we have the exact density of a one-orbital system then the potential is given by the formula

$$v_i = \frac{\phi_i'' + \varepsilon_i^{fd}}{2\phi_i}, \tag{A1}$$

where $\phi_i = \sqrt{n_i}$ and ε_i^{fd} is the error of the approximate second derivative at the point x_i. In the example shown in Figure A6 there are different error patterns ε_i^{fd} for the left- and right-hand sides of the box and this difference is magnified during the inversion because of the division by the square root of the density. A more realistic example includes some numerical error ε_i^n in the density being inverted and the inversion formula becomes

$$v_i = \frac{\left(\phi_i + \varepsilon_i^n\right)''}{2\left(\phi_i + \varepsilon_i^n\right)} = \frac{\phi_i'' + \varepsilon_i^{fd} + \left(\varepsilon_i^n\right)''}{2\left(\phi_i + \varepsilon_i^n\right)}. \tag{A2}$$

If the noise ε_i^n is not smooth then its second derivative will be very noisy and dominate the other terms in Equation A2. Figure A7 shows an example where a very small normally distributed error severely limits the accuracy of the potential.

Rounding errors

Rounding errors play an important role in many of the inversion methods dependent on numerical optimization methods. The Wu–Yang algorithm discussed in subsection 2.2 needs to compute the kinetic energy at each iteration of a given numerical optimization routine. If the density of a system decays exponentially then the kinetic energy from the asymptotic regions is very small in comparison to values near the nuclei. Even if the values of the kinetic energy density can be computed accurately in all regions of a simulation, adding all of the values together to get the total kinetic energy will often result in rounding errors due to the large differences in magnitude involved. In Figure A8, we apply the Wu–Yang inversion algorithm to the ground-state density of the harmonic oscillator on a grid with 51 points ranging from -8 to 8. Both the gradient-based TN algorithm[10] and the nongradient-based modified Powell's method[11] fail to resolve the density in the asymptotic region, ($|x|>5.5$), when used in the Wu–Yang inversion method. The abrupt cutoffs near -5.5 and 5.5 are a result of rounding errors when computing the kinetic energy with double-precision floating point numbers. If we use the midpoint integration rule to compute the kinetic energy in the region $|x|<5.5$ the result in double precision arithmetic is 0.2500000000000231 and the same method used in the region $|x|>5.5$ produces the value $-2.32460041955568\times10^{-14}$. When the two numbers above are added together, most of the smaller number's significant digits are lost and this loss of information limits the accuracy of the recovered potential in the asymptotic region. The same example performed using single-precision floating point numbers gives an even stricter cutoff near -3.5 and 3.5 as seen in Figure A9. In this second example the kinetic energy in the region $|x|<3.5$ is 0.25000963 when using single-point precision numbers compared to a value of -9.6271506×10^{-6} in the region $|x|>3.5$, which again results in a loss of significant figures when added together.

The preceding examples of time-independent density-to-potential inversions illustrate many of the pitfalls in performing numerical density-to-potential inversions. Simply refining the grid or increasing the accuracy of the kinetic energy operator does not guarantee an improved density-to-potential inversion and in many cases will worsen the solution. In cases where the error pattern is not known explicitly it is actually not correct to fit the density exactly as even small density errors can lead to large artifacts in the potential. Finally, inversions reliant on numerical optimization routines often suffer from rounding errors, which set yet another hard limit on the convergence of density-to-potential inversions.

APPENDIX B

DISCRETE-ADJOINT METHOD

In this appendix, we derive the discrete adjoint equations for the PDE-constrained optimization method presented in subsection 2.2.3. The discrete-adjoint method is a very efficient method for computing derivatives, (discrete functional derivatives), in PDE-constrained problems.[40] Computationally, these derivatives amount to a sparse linear solve for each occupied orbital and are about the same computational cost as computing the cost functional. The derivations give here are only valid for systems without degeneracies because this method is based on the algebraic method described in Ref. [41]. A similar derivation can be performed for systems with degeneracies using the method of Ref. [42].

TABLE 2 The notation used in the discrete-adjoint derivations

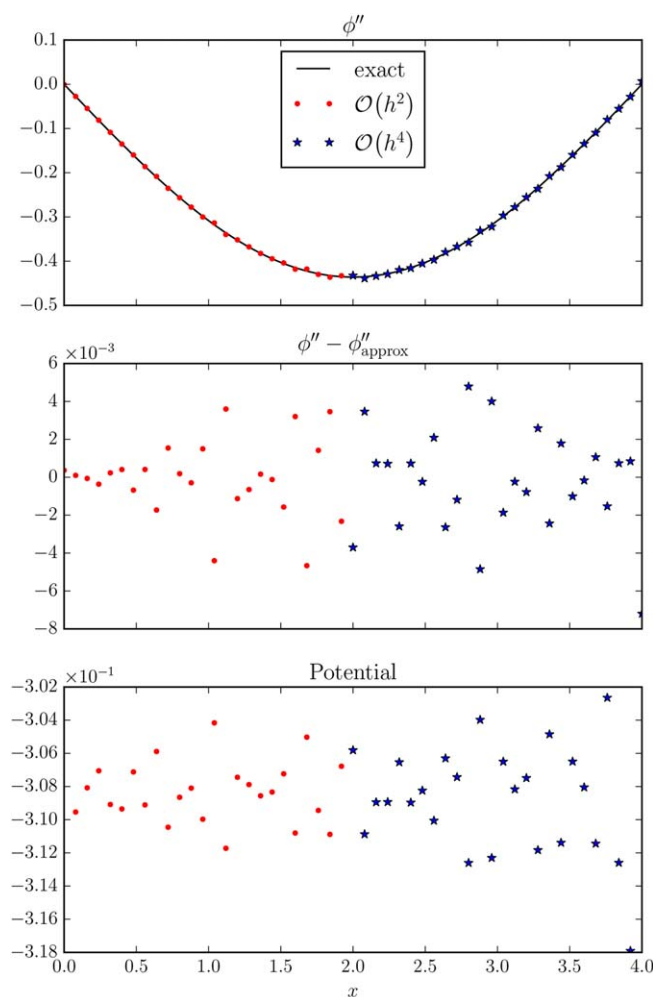| Notation	Description
$\mathbf{i}$	Discrete spatial indices i_x, i_y, i_z
$\sum_{i=1}^{N_x}$	Sum over all spatial indices $\sum_{i_x=1}^{N_x}\sum_{i_y=1}^{N_y}\sum_{i_z=1}^{N_z}$
F_n	Cost functional
$\mathcal{L}$	Lagrangian
$\tilde{n}$	Target density
N_{orbs}	Number of Kohn–Sham orbitals
ϕ^m and χ^m	mth orbital and corresponding adjoint orbital
$\underline{\phi}$	All N_{orbs} orbitals grouped into a vector
w_i	Optional cost-functional weight
dx_i and dk_j	Integration weights

FIGURE A7 The effects of combining two different finite difference operators in a density-to-potential inversion with a noisy density. This example is identical to the one shown in Figure A6 except the input density contains noise, (i.e., $\tilde{n}_0 = \{\phi_0 [1 + N (\mu = 0, \sigma^2 = 1 \times 10^{-10})]\}^2$). This is another example where higher-order methods can actually produce worse results than simpler lower-order methods. The scales on the y-axis of the middle and bottom plots are 10^{-3} and 10^{-1}, respectively

The notation used in the following derivations is listed in Table 2 and is the same notation used in Ref. [43] for TDDFT inverse problems. It is very similar to the continuous adjoint derivation found in quantum optimal control derivations applied to TDDFT.[44] The discrete adjoint method is discussed in depth in Ref. [45]. The notation used here is simply a modification of other discrete adjoint derivations applied to the DFT inverse problem.

The discretized cost functional

$$F[v] = F\left[\underline{\phi}[v]\right] = \frac{1}{2} \sum_{i=1}^{N_x} \left(\sum_{m=1}^{N_{orbs}} \phi_i^{m*} \phi_i^m - \tilde{n}_i \right)^2 w_i \tag{B1}$$

can be differentiated with respect to an arbitrary orbital ϕ_α^γ to produce the expression

$$\frac{\partial F}{\partial \phi_\alpha^\gamma} = w_\alpha \left(\sum_{m=1}^{N_{orbs}} \phi_\alpha^{m*} \phi_\alpha^m - \tilde{n}_i \right) \phi_\alpha^{\gamma*}, \tag{B2}$$

where γ is the orbital index and $\alpha = \alpha_x, \alpha_y, \alpha_z$ is the spatial index of the orbital being differentiated. We introduce the Lagrangian

$$\mathcal{L}\left[v, \underline{\phi}, \underline{\chi}\right] = 2\text{Re}\left\{ \sum_{i=1}^{N_x} \sum_{m=1}^{N_{orbs}} \chi_i^m [H(v)\phi^m - \varepsilon^m \phi^m]_i^* \right\} + \sum_{m=1}^{N_{orbs}} \lambda^m \left[(\phi_i^m)^* \phi_i^m - \tilde{n}_i \right] dx_i. \tag{B3}$$

where the Lagrange multipliers $\{\chi^m\}$ and $\{\lambda^m\}$ constrain the orbitals to satisfy Equations 4a and 3. The Hamiltonian $H(v) = -\frac{\Delta}{2} + v$ in Equation B3 is a discretized version of Equation 4a. We subtract the Lagrangian from the cost functional to form the total functional
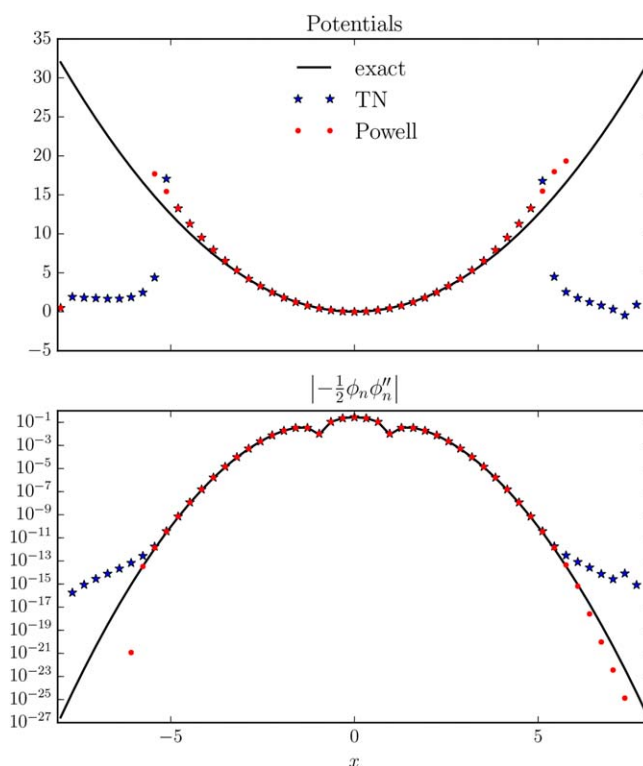
FIGURE A8 The effects of rounding errors in a density-to-potential inversion with double-precision numbers. Both the TN and modified Powell's optimizers fail to resolve the potential in the region $|x|>5.5$ (top) when used in the Wu–Yang algorithm and an initial guess of zero. The kinetic energy density is orders of magnitude smaller in the asymptotic region than it is in the region $|x|<5.5$ (bottom) and this leads to loss of precision when computing the total kinetic energy

$$J\left[v,\underline{\phi},\underline{\chi}\right]=F\left[\underline{\phi}\right]-\mathcal{L}\left[v,\underline{\phi},\underline{\chi}\right]. \tag{B4}$$

Differentiating the total functional J with respect to the adjoint orbitals $\{\chi^m\}$ and normalization multipliers $\{\lambda^m\}$ gives the standard KS equations

$$0=\frac{\partial J}{\partial \chi_\alpha^\gamma}\ \forall \alpha,\gamma\ \Rightarrow\ H(v)\phi^m=\varepsilon^m\phi^m\ \text{and} \tag{B5}$$

$$0=\frac{\partial J}{\partial \varepsilon^\gamma}\ \forall \gamma\ \Rightarrow\ \sum_{i=1}^{N_x}\left(\phi_i^m\right)^*\phi_i^m\cdot dx_i=1. \tag{B6}$$

The adjoint equations are found by differentiating with respect to the KS orbitals and eigenvalues to produce a set of sparse linear equations

$$\begin{bmatrix}\left(\frac{\partial F}{\partial \phi^\gamma}\right)^*\\\left(\frac{\partial F}{\partial \varepsilon^\gamma}\right)^*\end{bmatrix}=\begin{bmatrix}H^\dagger(v)-(\varepsilon^\gamma)^* & \phi^\gamma\\ -(\phi^\gamma)^\dagger & 0\end{bmatrix}\begin{bmatrix}\chi^\gamma\\\lambda^\gamma\end{bmatrix}, \tag{B7}$$

where $\frac{\partial F}{\partial \varepsilon^\gamma}=0$. After solving the direct and adjoint equations in Equations B5–B7, the total derivative is given by the expression

$$\frac{\partial J}{\partial v_\alpha}=-\frac{\partial \mathcal{L}}{\partial v_\alpha}=-2\sum_{m=1}^{N_{orbs}}\sum_{i=1}^{N_x}\text{Re}\{(\chi_i^m)^*\frac{\partial H(v)}{\partial v_\alpha}\phi_i^m\}. \tag{B8}$$

If the potential is placed diagonally along H then this last expression reduces to

$$\frac{\partial J}{\partial v_\alpha}=-2\sum_{m=1}^{N_{orbs}}\text{Re}\{(\chi_\alpha^m)^*\phi_\alpha^m\}. \tag{B9}$$

A very similar derivation is found when using the scaled orbitals described in subsection 2.2.3. In this case, the discretized cost functional is
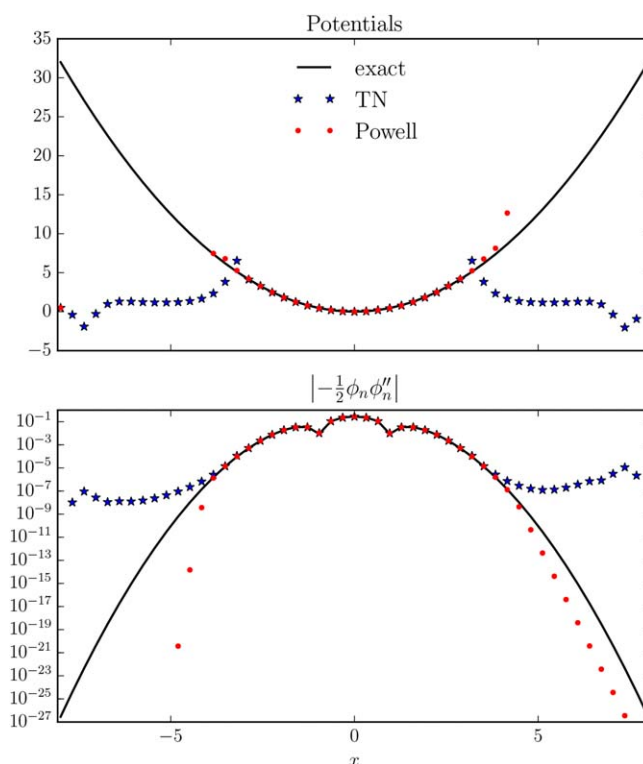
FIGURE A9 The effects of rounding errors in a density-to-potential inversion with single-precision numbers. This example is identical to the one shown in Figure A8 except that the kinetic energy sum uses single-precision numbers. In this case the potential is only recovered in the region $|x| < 3.5$ due to more severe rounding errors from the use of single-precision numbers

$$F[v] = F\left[\underline{g}[v]\right] = \frac{1}{2}\sum_{i=1}^{N_x} \left(\sum_{m=1}^{N_{orbs}} |g_i^m|^2 - 1\right)^2 w_i, \tag{B10}$$

where $\phi_i^m = \sqrt{\tilde{n}_i} g_i^m$ and

$$\frac{\partial F}{\partial g_\alpha^\gamma} = \left(\sum_{m=1}^{N_{orbs}} |g_\alpha^m|^2 - 1\right)\left(g_\alpha^\gamma\right)^* w_\alpha. \tag{B11}$$

The forward equations are

$$H(v)g^m = \varepsilon^m g^m \text{ and} \tag{B12}$$

$$\sum_{i=1}^{N_x} \tilde{n}_i g_i^{m*} g_i^m dx_i = 1. \tag{B13}$$

The corresponding adjoint equation is

$$\begin{bmatrix} \left(\frac{\partial F}{\partial g^\gamma}\right)^* \\ \left(\frac{\partial F}{\partial \varepsilon^\gamma}\right)^* \end{bmatrix} = \begin{bmatrix} H^\dagger(v) - (\varepsilon^\gamma)^* & \tilde{n}g^\gamma dx \\ -(g^\gamma)^\dagger & 0 \end{bmatrix}\begin{bmatrix} \chi^\gamma \\ \lambda^\gamma \end{bmatrix} \tag{B14}$$

and the total derivative is

$$\frac{\partial J}{\partial v_\alpha} = -2\sum_{m=1}^{N_{orbs}}\sum_{i=1}^{N_x} \mathrm{Re}\left\{(\chi_i^m)^* \frac{\partial H(v)}{\partial v_\alpha} g_i^m\right\}, \tag{B15}$$

where H is a discretization of the scaled Hamiltonian given in Equation 21.

The discrete adjoint equations for a periodic system follow the same pattern as the above derivations but with the added complication of a numerical integration in k-space. The discretized cost functional is

$$F[v]=F\left[\underline{\phi}[v]\right]=\frac{1}{2}\sum_{i=1}^{N_x} w_i\left(\sum_{m=1}^{N_{orbs}}\sum_{j=1}^{N_k} u_j|\phi_{i,j}^m|^2-\tilde{n}_i\right)^2,$$ (B16)

where $u_j=\frac{1}{\Omega_k}dk_j$ is the jth integration weight in k-space divided by the volume of the first Brillouin zone and $\phi_{i,j}^m$ corresponds to the point $\phi_{k_j}^m(x_i)$. The forward equations are

$$H_j(v)\phi_j^m=\varepsilon_j^m\phi_j^m \text{ and}$$ (B17)

$$\sum_{i=1}^{N_x}\left(\phi_{i,j}^m\right)^*\phi_{i,j}^m dx_i=1,$$ (B18)

where $H_j(v)$ is a discretization of the Hamiltonian $-\frac{1}{2}\left(\Delta+2\imath k_j\cdot\nabla-k_j^2\right)+v(\mathbf{r})$. The corresponding adjoint equations are

$$\begin{bmatrix}\left(\dfrac{\partial F}{\partial\phi_{\alpha,\beta}^\gamma}\right)^*\\[2ex]\left(\dfrac{\partial F}{\partial\varepsilon_\beta^\gamma}\right)^*\end{bmatrix}=\begin{bmatrix}(H_\beta)^\dagger(v)-\left(\varepsilon_\beta^\gamma\right)^* & \phi_{,\beta}^\gamma dx\\[2ex]-\left(\phi_{,\beta}^\gamma\right)^\dagger & 0\end{bmatrix}\begin{bmatrix}\chi_{,\beta}^\gamma\\[2ex]\lambda_\beta^\gamma\end{bmatrix}$$ (B19)

and the total derivative is

$$\frac{\partial J}{\partial v_\alpha}=-\frac{\partial\mathcal{L}}{\partial v_\alpha}=-2\sum_{j=1}^{N_k}\sum_{m=1}^{N_{orbs}}\sum_{i=1}^{N_x}\text{Re}\left\{\left(\chi_{i,j}^m\right)^*\frac{\partial H_j(v)}{\partial v_\alpha}\phi_{i,j}^m\right\}.$$ (B20)