**Problem**

**Solving the Kohn-Sham Equations**

\[ H[n]x_i = \lambda_i S x_i , \quad n = \sum_i |x_i|^2 \]

Where \( H[n] \) is the typical Kohn-Sham Hamiltonian with (for our work here) spinless LDA approximation, \( x_i, \lambda_i \) are eigenvector/eigenvalue pairs, \( n \) is the electron density, and \( S \) is the mass matrix resulting from a finite elements discretization.

**Traditional SCF**

The Kohn-Sham equations are traditionally solved using self-consistent field methods that follow the general procedure:

1. Find good initial guess \( n_1 \) (e.g. with Thomas-Fermi model)
2. Get \( n' = f(n_i) \), where \( f(n_i) \) is the procedure consisting of:
   a. Forming the Kohn-Sham hamiltonian \( H[n_i] \) by solving poisson’s equation, calculating the exchange-correlation operator, etc.
   b. Solving the eigenvalue problem \( H[n_i]x_i = \lambda_i S x_i \)
   c. Calculating the new density from the occupied states \( x_i \)
3. Form \( n_{i+1} \) from some function of \( n' \) and \( n_i \). If \( \|n_{i+1} - f(n_{i+1})\| < \epsilon \) then stop; otherwise GOTO step 2.

**Simple Mixing:**

\[ n_{i+1} = \alpha n_i + (1 - \alpha)n' , \quad 0 \leq \alpha \leq 1 \]

The value of \( \alpha \) is determined heuristically. Typical \( \alpha = 0.1 \). Convergence is usually very slow.

**DIIS:**

A list of \( m \) densities is first produced using some other procedure (e.g. simple mixing), and then subsequent \( n_{i+1} \) are given by[2]:

\[ n_{i+1} = \sum_{j=1}^{m} c_i n_{i-j} + c_i n_{i-j} \quad r_i = n_i - f(n_i) \]

where \( c_i \) are the coefficients that minimize \( \| r_i \| \) such that \( \sum c_j = 1 \). Convergence is known to be super linear.

**Nonlinear Eigensolver**

Our method is based on the FEAST algorithm[1], a method for solving linear eigenvalue problems. The nonlinear Kohn-Sham equations are solved in an approximate subspace \( Q \), which is updated at each iteration:

1. Get initial guess for \( X \), the matrix whose columns are the eigenvectors of interest \( x_i \), and \( n \), either or both of which can be completely random
2. Get density matrix \( \rho \) via numerical complex contour integration:
   \[ \rho = \int_{\mathbb{C}} (zI - H[n])^{-1} dz \]
   where \( C \) is a contour around the part of the real axis where the eigenvalues of the occupied states are expected to be found.
3. Form the subspace \( Q = \rho S X \)
4. Solve the projected, reduced nonlinear eigenvector problem
   \[ Q^T H[n]Qx_q = \lambda_q Q^T S Qx_q \]
   by iterating over the density \( n \). This procedure is computationally inexpensive compared to performing the full SCF iteration.
5. Calculate \( x_i = Qx_q \), \( \lambda_i = \lambda_q \), and the new density. If \( \sum \lambda_i \) has converged, exit. Otherwise, GOTO step 2.

**Advantages**

**Computational efficiency:** The primary computational costs of our method are mat-vec multiplication and boundary condition assignment in solving Poisson’s equation, both of which can easily be parallelized.

**No Initial Guess:** Our method converges on the correct result for arbitrary initial guess of density and subspace

**High Convergence rate:** Our method converges significantly faster than DIIS, and the convergence rate remains high regardless of system size.

**Future Work**

- Parallelize the solution of the subspace problem \( Q^T H[n]Qx_q = \lambda_q Q^T S Qx_q \). This will become the primary bottleneck for much larger systems.
- Update the FEAST package to provide a black-box interface for the algorithm described above. FEAST 2.0 currently provides black-box functionality for steps 2 and 3 described in “Nonlinear Eigensolver.”
- bgavin@ecs.umass.edu, epolizzi@ecs.umass.edu

**References**


**Results**

The following is the result of testing done with LDA all-electron DFT using a third order finite elements discretization. The total energy residual, the relative difference between the calculated total energy at subsequent iterations, is used as the means of gauging convergence.