Solution of large nonlinear eigenvalue problems
in Density Functional Theory

Yousef Saad

Department of Computer Science and Engineering

University of Minnesota

3rd Berlin-Manchester Workshop on Nonlinear Eigenvalue
Problems
March 22-23, 2007

Preliminary remarks

▶▶ **Increasing interest in problems related to condensed matter physics (analyzing properties of matter at the nanoscale).**

▶▶ **Problems lead to enormous computational challenges - excellent source of numerical problems of all types.**

▶▶ **Researchers in this area are among biggest users of high-performance computers**

▶▶ **Methods considered: ab-initio Density Functional Theory methods with Pseudopotentials.**

Electronic structure and Schrödinger's equation

▶▶ **The many-body Shrödinger equation:**

$$H\Psi = E\Psi$$

▶▶ **Hamiltonian H is of the form :**

$$H = -\sum_i \frac{\hbar^2 \nabla_i^2}{2M_i} - \sum_j \frac{\hbar^2 \nabla_j^2}{2m} + \frac{1}{2} \sum_{i,j} \frac{Z_i Z_j e^2}{|\vec{R}_i - \vec{R}_j|}$$
$$- \sum_{i,j} \frac{Z_i e^2}{|\vec{R}_i - \vec{r}_j|} + \frac{1}{2} \sum_{i,j} \frac{e^2}{|\vec{r}_i - \vec{r}_j|}$$

▶▶ $\Psi = \Psi(r_1, r_2, \ldots, r_n, R_1, R_2, \ldots, R_N)$ **depends on coordinates of all electrons/nuclei.**

Quick definition of the terms used:

Ab-initio methods – Methods which only use information about atoms that are present in the system

Density Functional Theory: Technique which expresses the original (intractable) problem into one with "charge density" as the main unknown

Pseudo-potential methods: Methods which model the nuclei-electrons interactions with a potential that 1) involves only valence electrons, 2) is smooth and nonsingular.

Density Function Theory - Kohn-Sham Eqns.

▶▶ Result of Density Functional Theory [Hohenberg-Kohn, Kohn-Sham]:

$$\left[-\tfrac{1}{2}\boldsymbol{\nabla}^2 + V_{tot}[\rho(r),r]\right]\Psi_k(r) = E_k\Psi_k(r)$$

With

$$V_{tot} = V_{ion} + V_H + V_{xc}$$

- V_H = Hartree potential   **local**
- V_{xc} = Exchange & Correlation potential   **local (LDA)**
- V_{ion} = Ionic potential   **Non-local**

▶▶ Electron Density:

$$\rho(r) = \Sigma_i^{occup}\,|\Psi_i(r)|^2$$

Kohn-Sham equations $\rightarrow$ nonlinear eigenvalue Pb

$$\left[-\frac{1}{2}\nabla^2 + V_{tot}[r, \rho(r)] \right] \Psi_i(r) = E_i \Psi_i(r), i = 1, ..., i^o$$

$$V_{tot} = V_H + V_{xc} + V_{ion}$$

$$\rho(r) = \overset{o}{\underset{i}{\Sigma}} |\Psi_i(r)|^2$$

$$\nabla^2 V_H = -4\pi\rho(r)$$

▶▶ Both V_{xc} and V_H, depend on ρ.

▶▶ Potentials & charge densities must be **self-consistent**. Can be viewed as a nonlinear eigenvalue problem

▶▶ Broyden-type quasi-Newton 'mixing' technique used

▶▶ Typically, a small number of iterations are required

Solution methods used

1. SCF + Optimization viewpoint: Minimize energy – To some extent amounts to minimizing trace.. [but Pb is nonlinear] – Car-Parrinello is in this category [simulated anneal.]

2. SCF + diagonalization viewpoint: solve eigenvalue problem at each SCF iteration.

3. Linear scaling methods – determine the density matrix = a projector whose diagonal equals ρ :

$$V = [\psi_1, \cdots, \psi_m] \ \rightarrow \ P = VV^T \ \rightarrow \ \rho = diag(P)$$
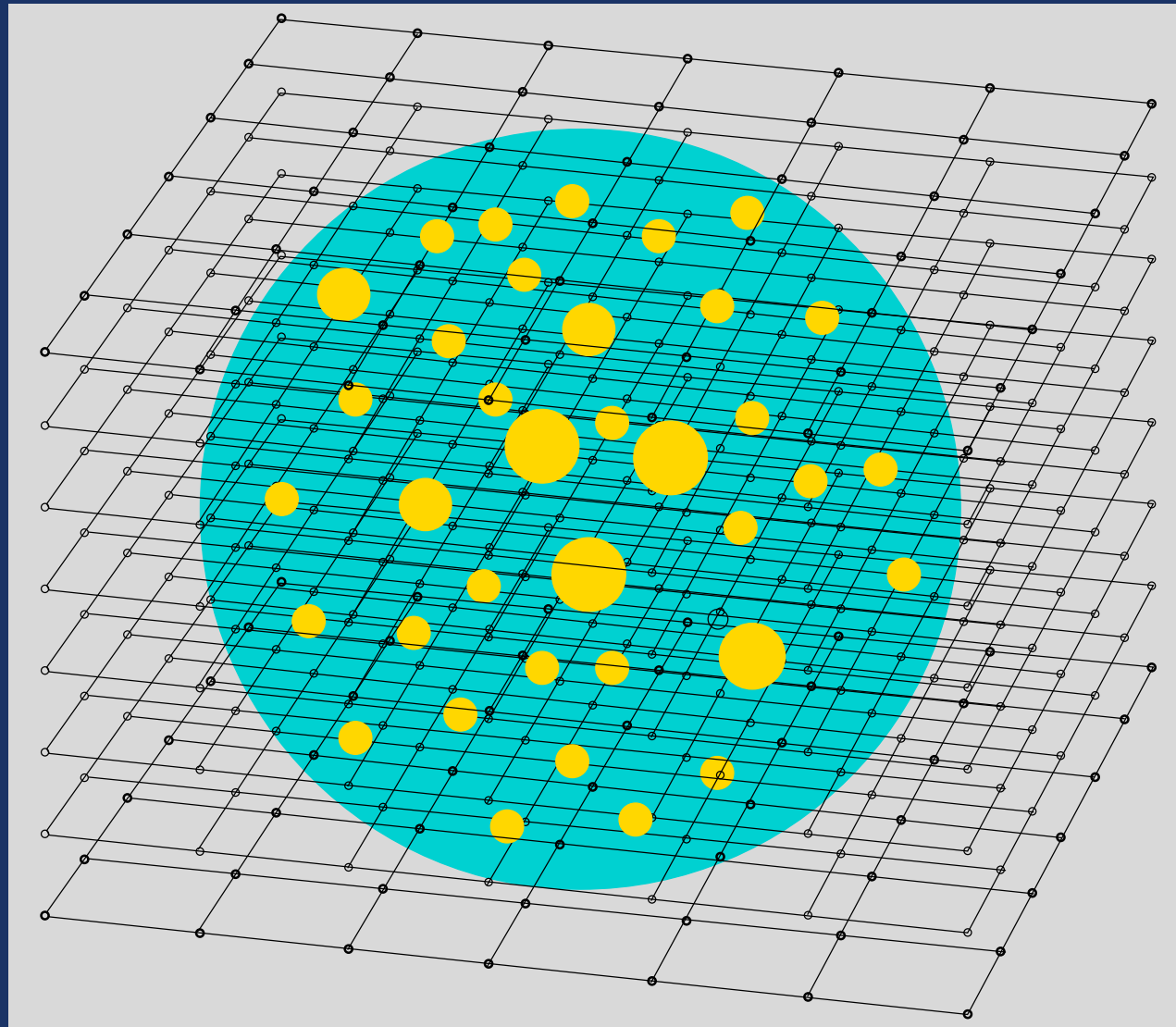
Real-space Finite Difference Methods

▶▶ **Use High-Order Finite Difference Methods [Fornberg & Sloan '94]**

▶▶ **Typical Geometry = Cube – regular structure.**
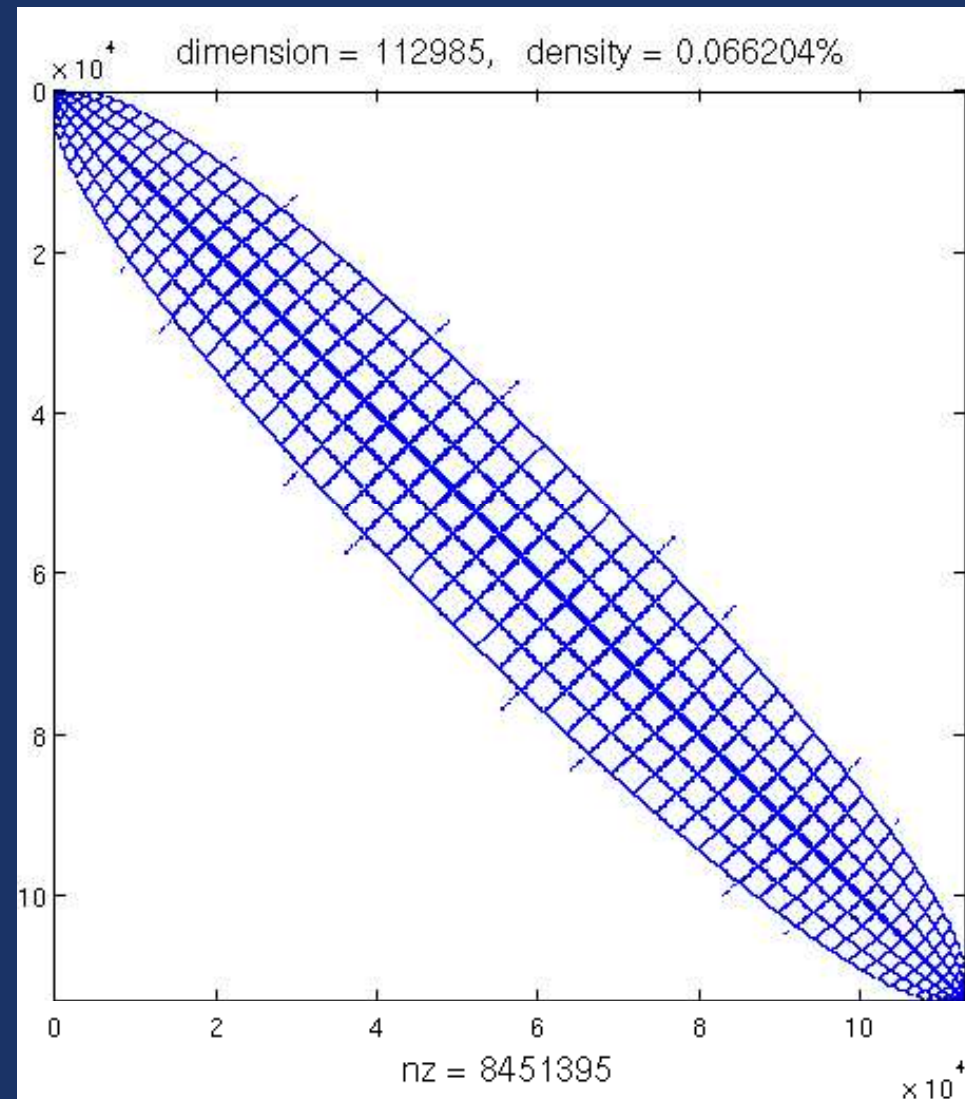
▶▶ **Laplacean matrix need not even be stored.**

Order 4 Finite Difference Approximation:
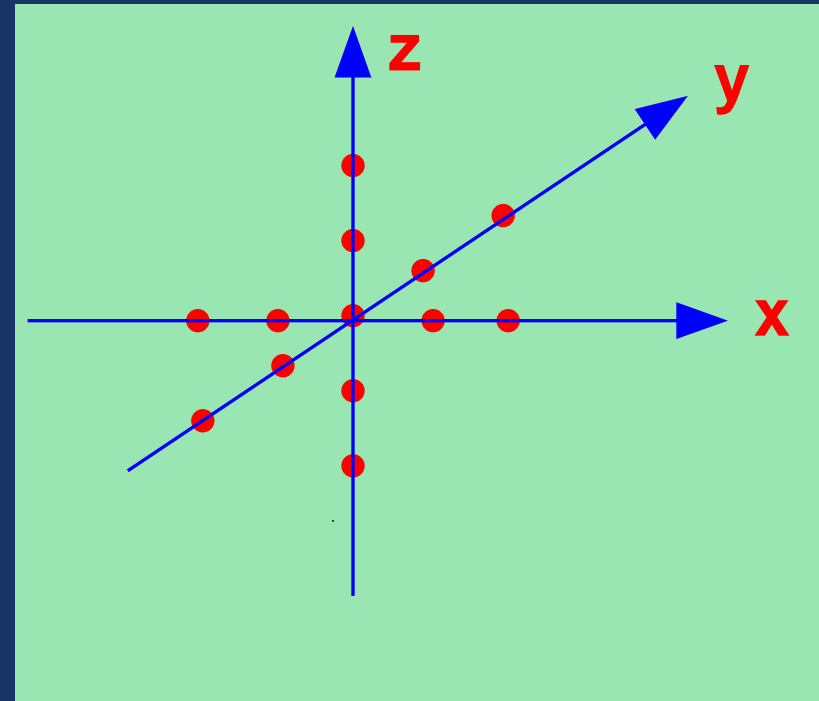
The physical domain

Pattern of resulting matrix for Ge99H100:

dimension = 112985, density = 0.066204%

nz = 8451395

Matlab version - RSDFT

```
function [rho, lam, W] = rsdft(A, nev, Domain, Atoms,
                              tol, maxits, fid)
%% IN:
%% A       = sparse matrix representing the discretization
%%           of the Laplacean --
%% nev     = number of eigenvalues = # of occupied states
%% Domain  = struct containing info on the physical domain
%% Atoms   = struct containing info on the atoms
%% tol     = tolerance parameter for  scf iteration.
%% maxits  = maximum number of SCF iterations allowed.
%% fid     = output file id
```

```
%% OUT    :
%% rho    = final charge density found
%% lam    = eigenvalues computed - their number may be larger
%% W      = set of wave functions.
%%=================================================================
```

A nonlinear form of Chebyshev subspace iteration

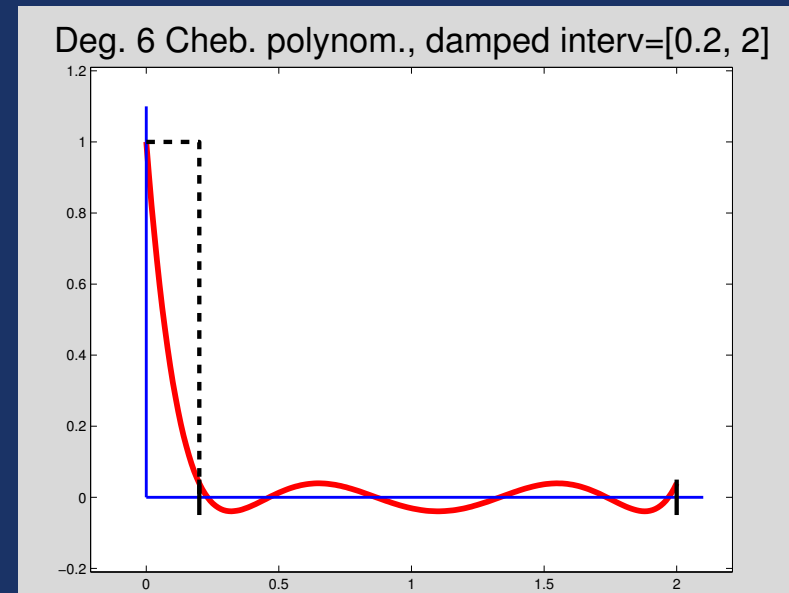Given a basis $[v_1, \ldots, v_m]$, 'filter' each vector as

$$\hat{v}_i = P_k(A)v_i$$

▶▶ p_k = Polynomial of low degree: enhances desired eigen-components

The **filtering step** is not used to compute eigenvectors accurately ▶▶

SCF & diagonization loops merged

Important: convergence still good and robust

Deg. 6 Cheb. polynom., damped interv=[0.2, 2]

Main step:

Previous basis $V = [v_1, v_2, \cdots, v_m]$

$\downarrow$

Filter $\qquad \hat{V} = [p(A)v_1, p(A)v_2, \cdots, p(A)v_m]$

$\downarrow$

Orthogonalize $[V, R] = qr(\hat{V}, 0)$

▶ The basis V is used to do a Ritz step (basis rotation)

$$C = V^T A V \rightarrow [U, D] = eig(C) \rightarrow V := V * U$$

▶ Update charge density this basis.

▶ Update Hamiltonian — repeat

▶ In effect: **Nonlinear subspace iteration**

▶ Main advantages: (1) very inexpensive, (2) uses minimal storage (m is a little $\geq$ # states).

▶ Filter polynomials: if $[a,\ b]$ is interval to dampen, then

$$p_k(t) = C_k(l(t)); \quad \text{with} \quad l(t) = \frac{2t - b - a}{b - a}$$

$+$ add scaling

▶ 3-term recurrence of Chebyshev polynommial exploited to compute $p_k(A)v$. If $B = l(A)$, then

$$p_{k+1}(A)v = 2Bp_k(B)v - p_{k-1}(B)v \rightarrow w_{k+1} = 2Bw_k - w_{k-1}$$
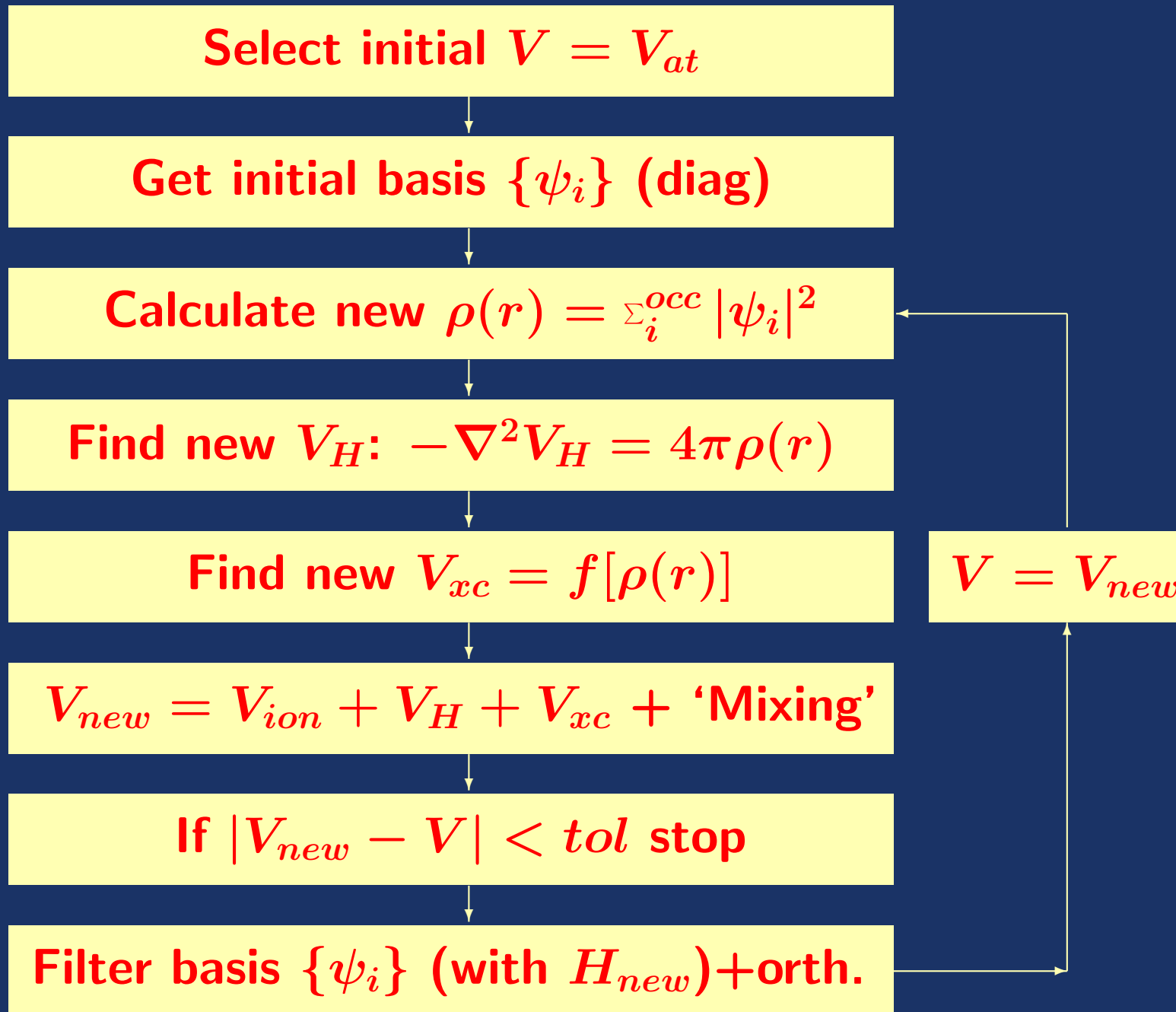
Reference:

Yunkai Zhou, Y.S., Murilo L. Tiago, and James R. Chelikowsky, **Self-Consistent-Field Calculations with Chebyshev Filtered Sub-space Iteration,** *Minnesota Supercomputer Institute, tech. report., Oct. 2005.*

[See http://www.cs.umn.edu/~saad]

Chebyshev Subspace iteration - experiments

model	size of H	n_{state}	n_{symm}	$n_{H-reduced}$
$Si_{525}H_{276}$	292,584	1194	4	73,146
$Si_{65}Ge_{65}H_{98}$	185,368	313	2	92,684
$Ga_{41}As_{41}H_{72}$	268,096	210	1	268,096
Fe_{27}	697,504	520×2	8	87,188
Fe_{51}	874,976	520×2	8	109,372

Test problems

▶▶ **Tests performed on an SGI Altix 3700 cluster (Minnesota supercomputing Institute). [CPU = a 1.3 GHz Intel Madison processor. Compiler: Intel FORTRAN `ifort`, with optimization flag `-O3`]**

method	# $A * x$	SCF its.	CPU(secs)
ChebSI	124761	11	5946.69
ARPACK	142047	10	62026.37
TRLan	145909	10	26852.84

$Si_{525}H_{276}$, **Polynomial degree used is 8. Total energies agreed to within 8 digits.**

method	# $A * x$	SCF its.	CPU (secs)
ChebSI	474773	37	37701.54
ARPACK	1272441	34	235662.96
TRLan	1241744	32	184580.33

Fe_{51}, **Polynomial degree used is 9. Total energies same to within ~ 5 digits.**

Some recent results - with *Large* Si clusters

$Si_{6047}H_{1308}$

n_{state}	# $A*x$	# SCF	$\frac{total\_eV}{atom}$	1st CPU	total CPU
12751	2682749	14	-91.34809	45.11 h.	101.02 h.

PEs = 32. n_H =2,144,432. $m = 17$ for Chebyshev-Davidson; $m = 8$ for CheFSI.

$Si_{9041}H_{1860}$

n_{state}	# $A*x$	# SCF	$\frac{total\_eV}{atom}$	1st CPU	total CPU
19015	4804488	18	-92.00412	102.12 h.	294.36 hrs

PEs = 48; n_H =2,992,832. $m = 17$ for Chebyshev-Davidson; $m = 8$ for CheFSI.

Summary & Conlusion

▶▶ **Very important to consider problem from the angle of 'subspaces' rather than individual eigenvectors.**

▶▶ **Also important: SCF loop. A 'sub-optimal' linear algorithm becomes a star in nonlinear context!**

▶▶ **Next big step: completely avoid diagonalization ['linear scaling' methods w. density matrix formalism]**

▶▶ **Use better filtering poly-nomials? Can exploit orthogonal polynomials. see [YS, 2005]**

▶▶ **It is possible to find good polynomials for spectrum slicing [computing eigenspaces by parts, independently]. Issue: implementation.**

Poly. filter; Intervals: [0   0.3] ; [0.3   1.3927] ; [1.3927   7.957] ; deg =

- **My URL:**

> **URL: http://www.cs.umn.edu/∼saad**

- **My e-mail address:**

> **e-mail: saad@cs.umn.edu**

- **PARSEC's site:**

> **http://www.ices.utexas.edu/parsec/index.html**

> **THANK YOU FOR YOUR ATTENTION!**