THE UNIVERSITY *of York*

Finding the electronic ground state

Matt Probert
Condensed Matter Dynamics Group
Department of Physics,
University of York, U.K.
http://www-users.york.ac.uk/~mijp1

Overview of lecture

- Exact diagonalization
- Iterative diagonalization
  - Indirect approach
  - Direct approach
- CASTEP in practice

- NB NB Colour slides are available on the website

Exact Diagonalization

Exact Diagonalization

- We want to solve

$$\mathrm{H}[\rho]\psi_{bk} = \epsilon_{bk}\psi_{bk}$$

- to find the eigenenergies ϵ_{bk} and $\psi_{bk}(\mathbf{r})$ eigenfunctions which we represent in terms of the coefficients c_{Gk}
- For a simple matrix problem **Hx**=λ**x** we can solve by *diagonalizing* **H**
  - BUT unlike simple problems we do not know the matrix elements of **H** before we start!

The trouble with H

- Note that H depends on V
  - And V is a functional V[ρ]
  - And ρ depends on ψ
  - And we are trying to diagonalize H to find ψ!
- Hence we must solve iteratively:
  - Guess an initial ρ –> V[ρ] -> H
  - Solve $H\psi=\varepsilon\psi$ to get new approximate ψ
  - Compute new ρ ; repeat to convergence …

Cost of Exact Diagonalization

- Given the above, we can make an $N_G x N_G$ matrix H and diagonalize it
  - Standard linear algebra packages can do this in $O(N_G^3)$ operations and $O(N_G^2)$ storage
  - And must repeat for each k-point **k**
  - And iterate to self-consistency
- How big is N_G?
  - Consider a 10Å^3 box and cut-off energy of 500 eV then $N_G \sim 50600$
  - Estimate time ~12 hrs/iteration at 3 GFLOPs

Trouble

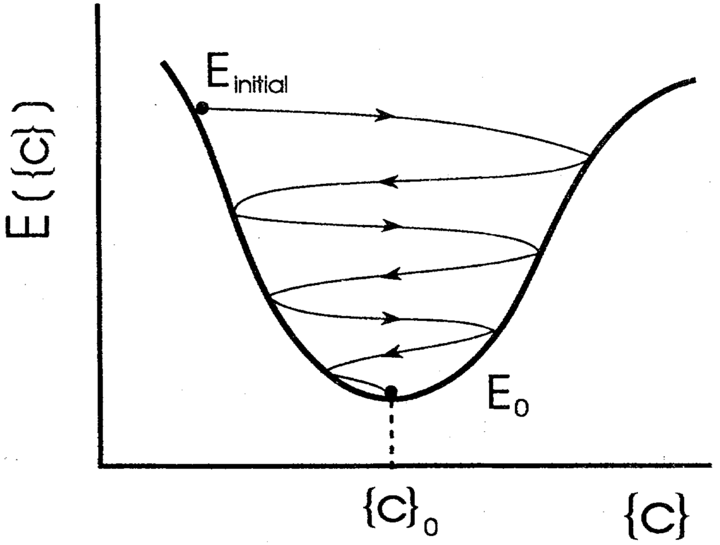
- Exact diagonalization is VERY expensive
  - And an $N_G x N_G$ matrix has N_G eigenvalues
  - But typically only want the lowest few eigenvalues ~ $N_{electrons}$
  - And takes a lot of computer power to get machine precision in all eigenvalues which have to do for every iteration even when far from self-consistency
- State of the art up until 1985
  - Limited DFT to a max of ~10 electrons!

Iterative Diagonalization
- indirect approach

Revolutionary new approach

- Car and Parrinello (1985)
  - Inspired by extended Lagrangian methods in Molecular Dynamics
  - Introduced fictitious ‘mass’ and ‘kinetic energy’ for each c_{Gk}

Do MD with damping in the space of c_{Gk} so as system ‘cools’ it converges to ground state value of c_{Gk} and hence electron density etc.



Car-Parrinello approach

- Major algorithmic break through!
- Cost ~$\mathcal{O}(N_G^2 N_b)$ to apply H to all bands
  - And explicit orthogonalization step costs ~$\mathcal{O}(N_G N_b^2)$
  - BUT $N_b \ll N_G$ …
  - And once got to ground state can combine with conventional MD of ions to get *ab initio* MD for the first time
- Still requires ~$\mathcal{O}(N_G^2)$ storage

Car-Parrinello Drawbacks

- But to ensure adiabatic separation of electrons and ions need mass separation
  - Choose very small ‘mass’ for c_{Gk}
- Hence need very small time step to integrate the equations of motion
  - Hence not actually that much faster than exact diagonalization in ‘time to science’
  - And cannot handle metals due to adiabatic issues

Energy Minimization

- One view of the C-P approach was that it could be seen as indirect energy minimization
  - But simulated annealing is VERY inefficient – better for global than local optimization
- So why not use a more efficient direct function minimization approach?
  - Conjugate-gradients introduced by Teter, Payne & Allan (1989) along with an efficient pre-conditioner
  - Objective: energy eigenvalues

Iterative Diagonalization
- direct approach

Direct minimization basics

- The groundstate energy E_0 is the lowest possible energy of the system
- Any wavefunction has energy $E \geq E_0$ so
  - Guess a trial wavefunction ψ
  - Compute $E = \psi^\dagger H \psi$
  - Tweak c_G to lower E
  - When we cannot lower E any more then ψ is the groundstate!
  - Variational principle in action

Iterative Diagonalization

- Efficient minimization methods need both objective function and its derivative
- Functional calculus time:

$$
\begin{aligned}
\epsilon_{bk} &= \frac{\psi_{bk}^{\dagger} \mathrm{H} \psi_{bk}}{\psi_{bk}^{\dagger} \psi_{bk}} \\
\Rightarrow \frac{\delta \epsilon_{bk}}{\delta \psi_{bk}^{\dagger}} &= \frac{\left(\psi_{bk}^{\dagger} \psi_{bk}\right) \mathrm{H} \psi_{bk} - \left(\psi_{bk}^{\dagger} \mathrm{H} \psi_{bk}\right) \psi_{bk}}{\left(\psi_{bk}^{\dagger} \psi_{bk}\right)^{2}} \\
&= \mathrm{H} \psi_{bk} - \epsilon_{bk} \psi_{bk}
\end{aligned}
$$

- And so gradient vanishes at minimum …

Steepest Descents

- Start with ψ_j and compute ϵ_j
- Compute the gradient $\frac{\delta\epsilon_j}{\delta\psi_j^\dagger}$
  - This is the change to c_G that increases ϵ_j
  - We want to decrease ϵ_j so use negative
- Guess new eigenstate $\psi_j^{new} = \psi_j - \lambda\frac{\delta\epsilon_j}{\delta\psi_j^\dagger}$
- Vary λ until found min ϵ_j in this direction
  - This is called the *line minimization* step
- Update, recompute gradient, repeat …

Multiple eigenstates

- If apply this procedure then will quickly find lowest eigenstate ψ_1
- Repeating it for ψ_2 will give same answer!
- The solution is to explicitly orthogonalize so

$$\psi_2^\dagger \psi_1 = 0$$

- So need good matrix algebra package/code for multiplication, orthogonalization ….

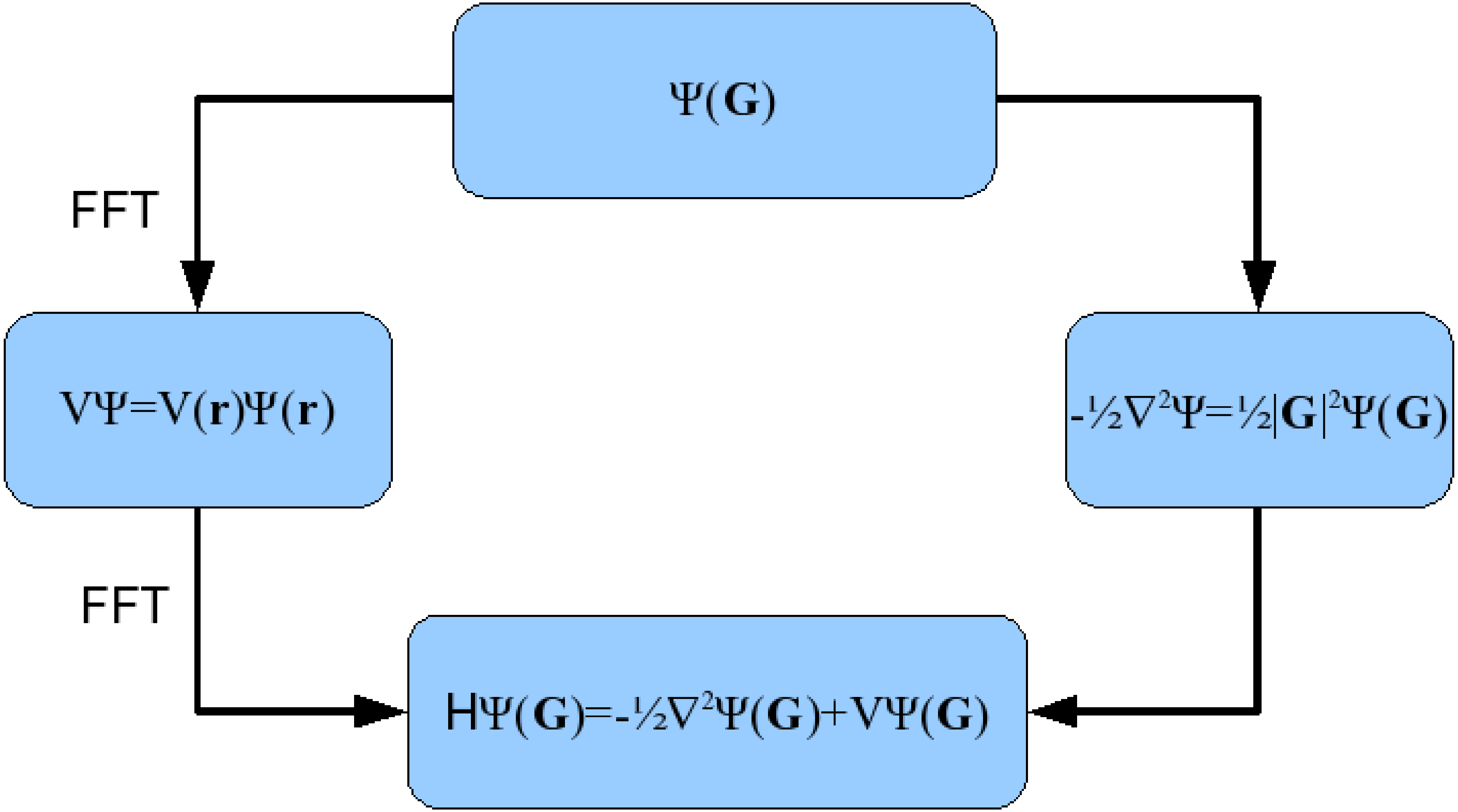
Efficient application of H

- Remember:

$$\hat{H} \;=\; -\frac{\hbar^2}{2m}\nabla^2 + V\left[\rho\right](\mathbf{r})$$

- So can split into two terms:
- Kinetic energy – trivial to apply in reciprocal space: $\nabla^2\psi_{bk}(\mathbf{G}) = -|\mathbf{G}+\mathbf{k}|^2\psi_{bk}(\mathbf{G})$
- Potential energy – trivial to apply in real space: $V\left[\rho\right](\mathbf{r})\psi_{bk}(\mathbf{r})$
- Hence use Fast Fourier Transforms to switch spaces …

Applying the Hamiltonian



$\Psi(\mathbf{G})$

FFT

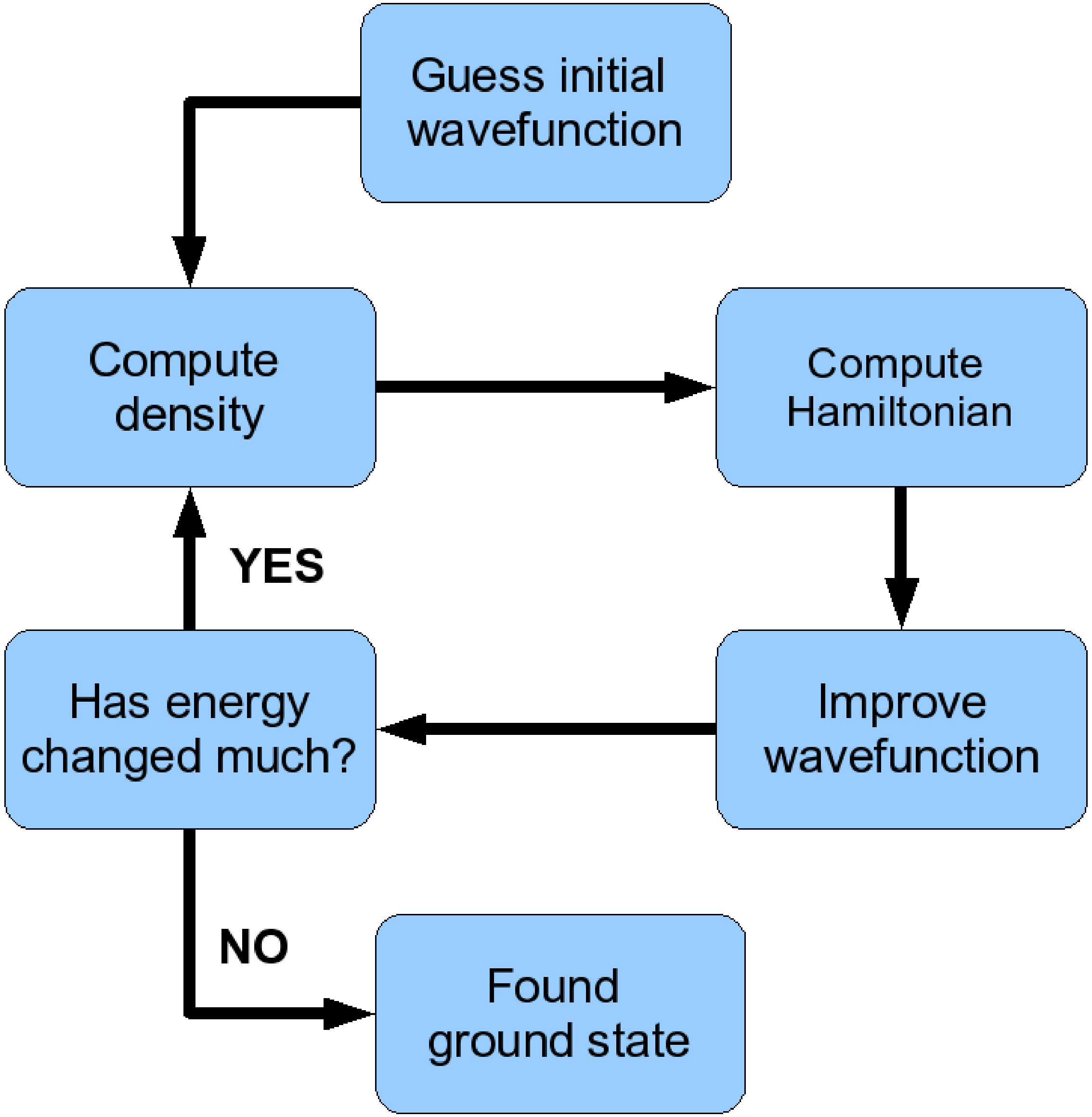
$V\Psi = V(\mathbf{r})\Psi(\mathbf{r})$

$-\frac{1}{2}\nabla^2\Psi = \frac{1}{2}|\mathbf{G}|^2\Psi(\mathbf{G})$

FFT

$H\Psi(\mathbf{G}) = -\frac{1}{2}\nabla^2\Psi(\mathbf{G}) + V\Psi(\mathbf{G})$

Finding the Groundstate



Guess initial wavefunction

Compute density

Compute Hamiltonian

Improve wavefunction

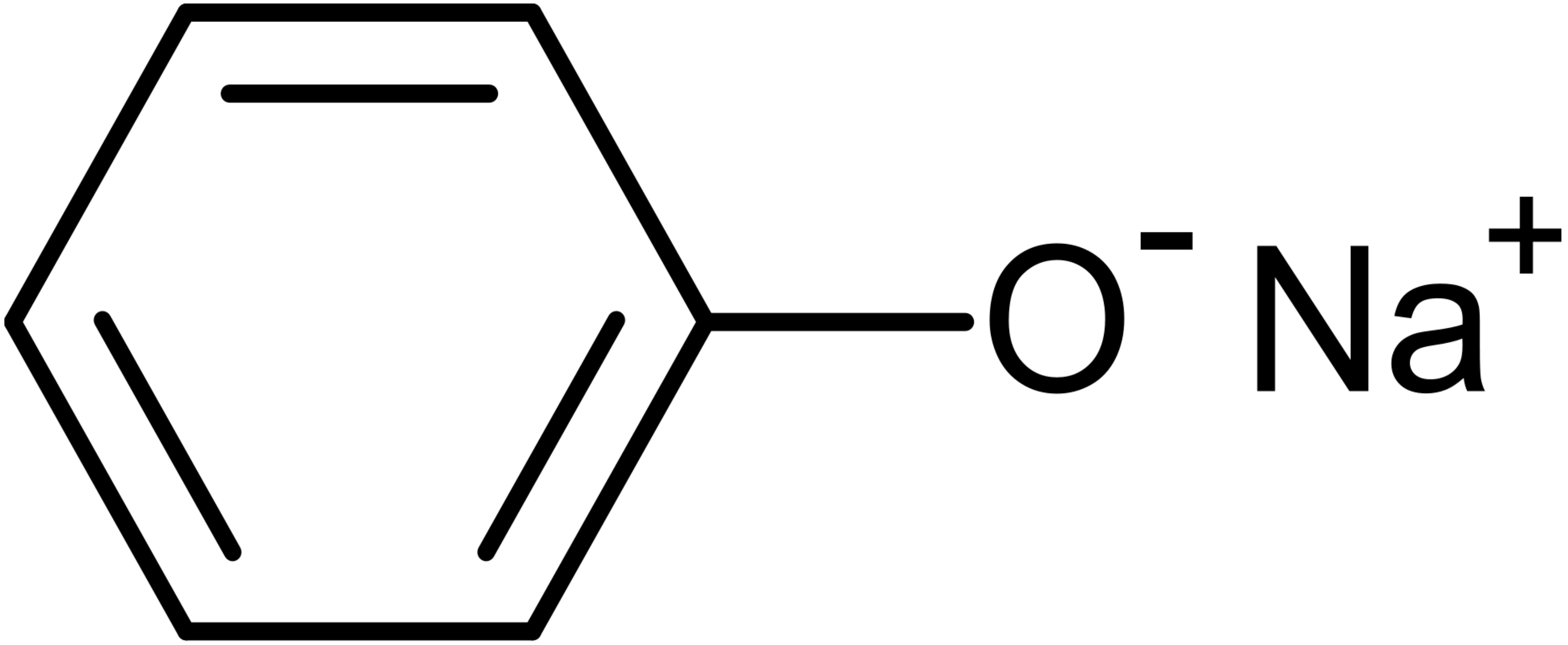
Has energy changed much?

Found ground state

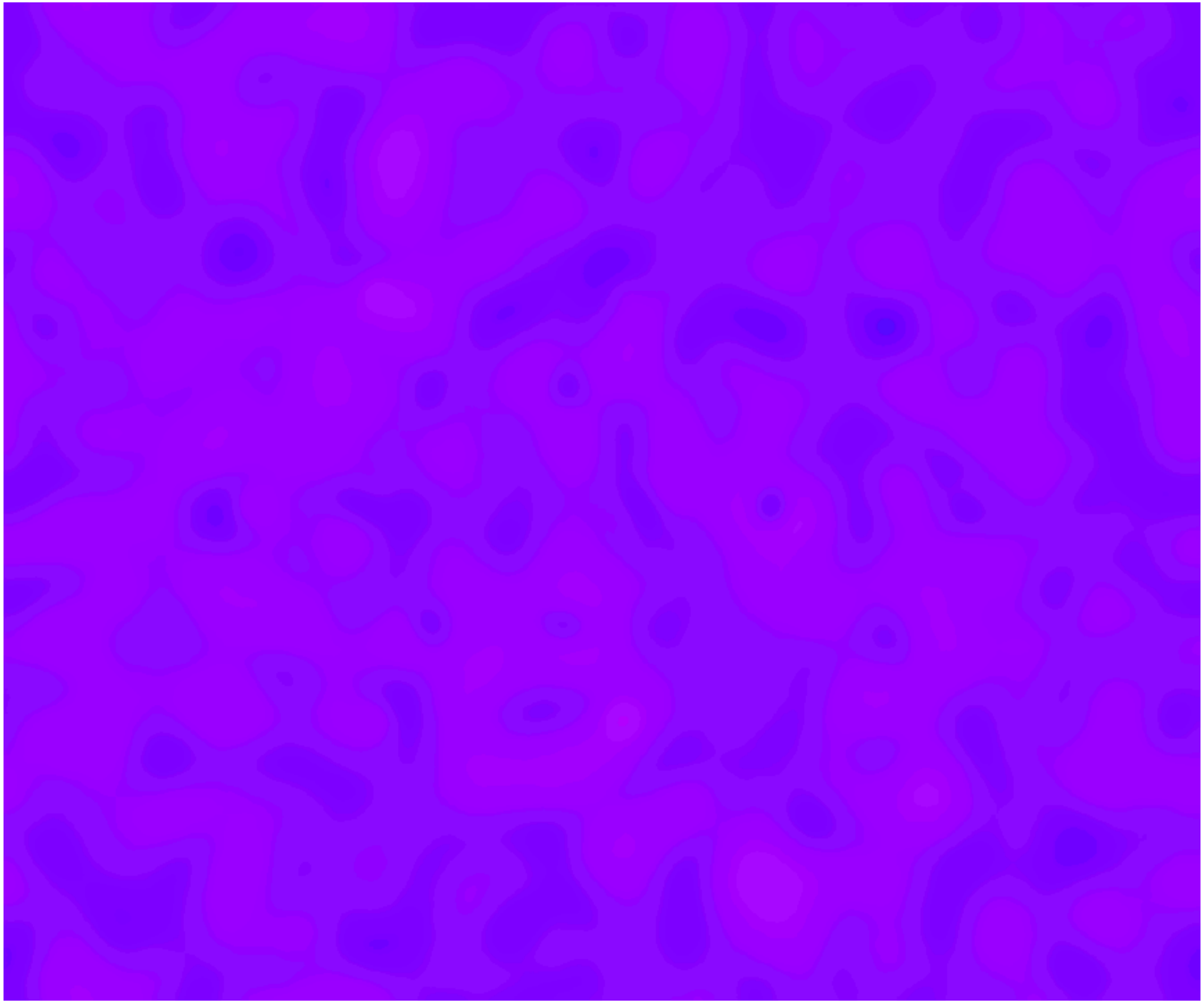
Conjugate Gradients

- Steepest descents works – robust but slow
- Conjugate gradients is more efficient
- Both schemes have similar advantages:
  - Never need to store H explicitly
    - Hence storage ~$O(N_G)$
  - Smart use of real/reciprocal space means cost of applying H ~$O(N_G N_B)$
  - Orthogonalization of bands ~$O(N_G N_B^2)$
  - Cost of FFT ~$O(N_G N_B \ln N_G)$

Minimization in action



Minimization in action



Direct Minimization Advantages

- Direct minimization does not need adiabatic separation of electrons and ions
  - Hence handles metallic states easily
- Once got ground state can use Hellman-Feynman theorem to get forces and hence do Born-Oppenheimer MD
  - With a much bigger time step than C-P
  - Traditionally had worse energy conservation than C-P but no longer true with recent developments such as XL-BOMD

CASTEP in practice

Self consistency

$$\epsilon_{b\mathbf{k}}^{new} = \langle \psi_{b\mathbf{k}}^{new} \,|\, \hat{H} \,|\, \psi_{b\mathbf{k}}^{new} \rangle$$

- But *H* depends on ρ so should we use old ρ or new ρ?

$$\rho^{new} = \sum_{b\mathbf{k}} |\psi_{b\mathbf{k}}^{new}|^2$$

- Does it matter?

Density mixing

- Updating ρ requires Fourier transform of every band → correct but slow
  - Or can fix $\rho = \rho_{in}$ and optimise ψ so no longer self-consistent until at convergence when $\rho_{\text{out}} = \rho_{in}$
  - This is known as *density mixing* (DM)
  - Simplest scheme is linear mixing:

$$\rho = (1-\alpha)\rho^{in} + \alpha\rho^{out}$$

- DM is fast and requires less memory but is sometimes unstable …

Charge Sloshing

- Problems arise chiefly due to Hartree potential (classical electron-electron repulsion):

$$V_H(\mathbf{G}) \propto \frac{\rho(\mathbf{G})}{|\mathbf{G}|^2}$$

  - Small errors in ρ_{in} → large errors in $V_H[\rho_{in}]$ → large errors in ρ_{out} .
  - This phenomenon is called *charge sloshing*.
- Solution is to mix densities to correct for this viz. Kerker mixing:

$$\rho^{new}(\mathbf{G}) = \rho^{in}(\mathbf{G}) + \frac{\alpha|\mathbf{G}|^2}{|\mathbf{G}|^2+G_0^2}\left(\rho^{out}(\mathbf{G}) - \rho^{in}(\mathbf{G})\right)$$

Advanced DM

- Better still – model the dielectric response:

$$\rho^{new}\left(\mathbf{G}\right)=\rho^{in}\left(\mathbf{G}\right)+\epsilon_{\mathbf{GG'}}^{-1}\left(\rho^{out}\left(\mathbf{G'}\right)-\rho^{in}\left(\mathbf{G'}\right)\right)$$

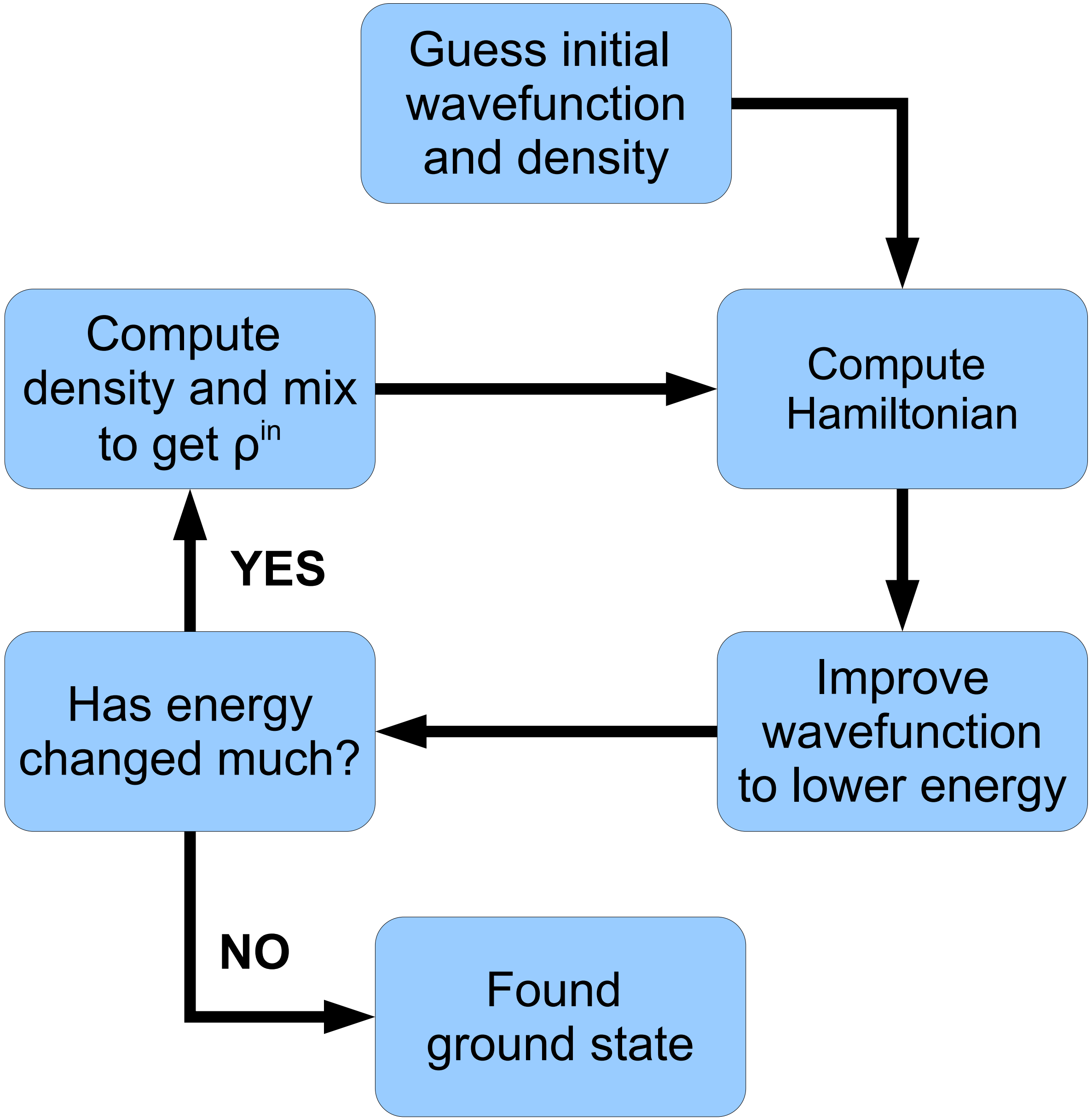
- Where $\epsilon_{\mathbf{GG'}}^{-1}=1-\frac{\delta\rho\left(G\right)}{\delta\rho\left(G'\right)}$

- Different schemes (Pulay or Broyden) available - both start with

$$\epsilon_{\mathbf{GG'}}^{-1}\simeq\frac{\alpha\left|\mathbf{G}\right|^{2}\delta_{GG'}}{\left|\mathbf{G}\right|^{2}+G_{o}^{2}}$$

- And then improve $\varepsilon^{-1}{}_{GG'}$ using a history of mixing densities

DM scheme



Guess initial wavefunction and density

Compute Hamiltonian

Improve wavefunction to lower energy

Has energy changed much?

YES

Compute density and mix to get ρ^{in}

NO

Found ground state

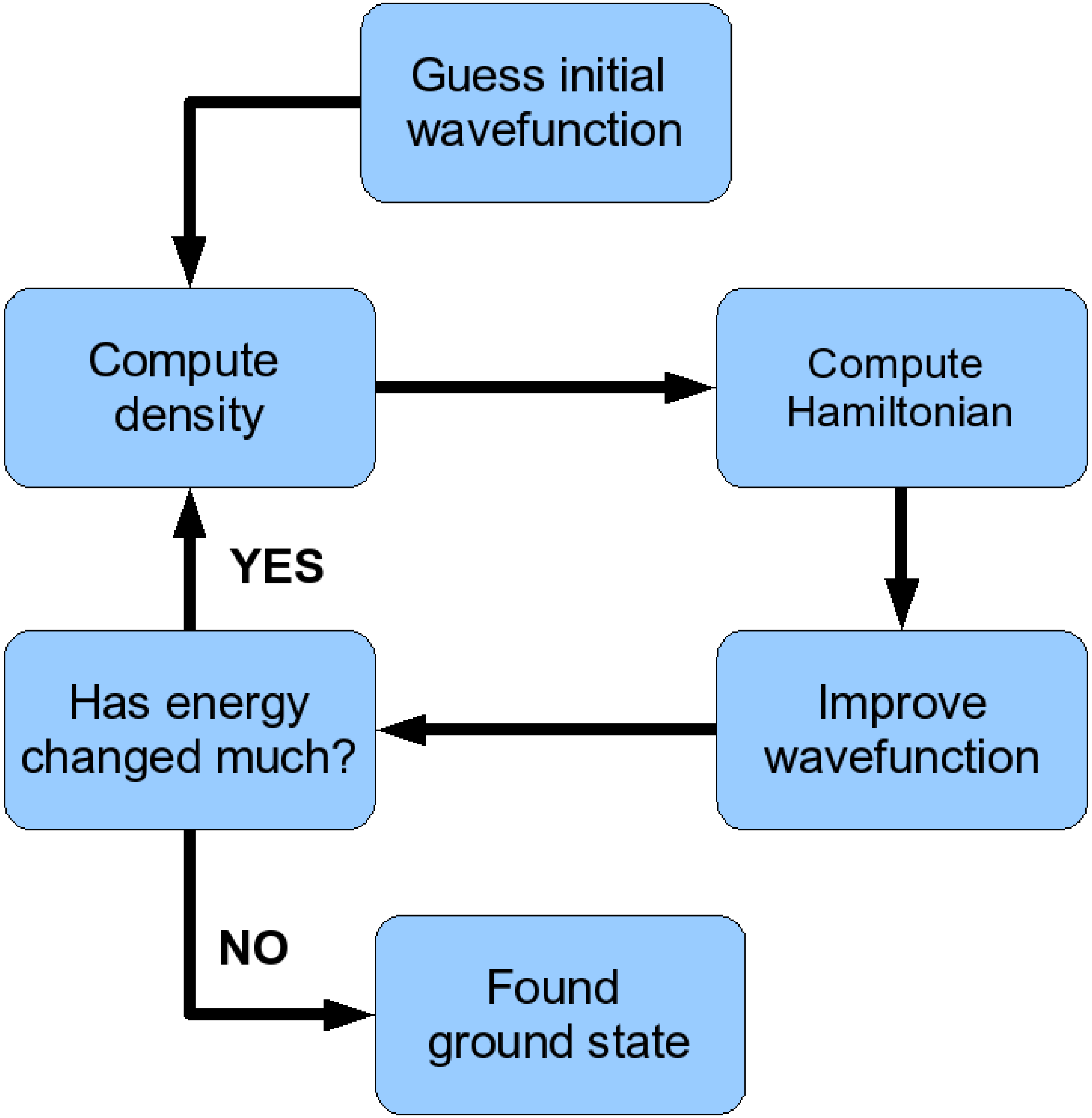
DM difficulties

- Sometimes DM does not converge
- Can try different CASTEP parameters:
  - Try a different DM scheme in `.param` file: **`dm_mix_scheme = linear / Kerker / Pulay / Broyden`**
  - Reduce **`mix_charge_amp`** to 0.1~0.2
  - Increase **`mix_cut_off_energy`** (up to 4***`cut_off_energy`**)
  - Increase **`mix_history_length`**
- Or switch to a non-DM scheme …

EDFT

- Always update ρ so fully self-consistent but computational expensive:

```
metals_method = EDFT
```



Guess initial wavefunction

Compute density

Compute Hamiltonian

Improve wavefunction

Has energy changed much?

YES

NO

Found ground state

Metal or Insulator?

- Everything up to now has been for insulators
- Metals have degenerate states at $E=E_F$
  - Problems at T=0 due to occupancy discontinuity
  - Solution – run at finite T and smear
  - Fractional occupancies:

$$\rho = \sum_{bk} f_{bk} |\psi_{bk}|^2 \; ; \; E = \sum_{bk} f_{bk} \epsilon_{bk}$$

  - Need extra bands and care with k-points!

Summary

Summary

- DFT reduces QM to a matrix e-value problem
  - Where number of useful e-states is SMALL compared to size of matrix
  - Hence iterative diagonalization is best
- DM is a fast scheme but not always stable:
  - Fix ρ when update ψ
  - Energy converges faster than forces
- EDFT is slower but stable
  - Always update ρ when update ψ
  - Energy and forces converges fast

Useful References

- MC Payne *et al*, Rev. Mod. Phys **64**, 1045 (1992)
- WH Press *et al*, "*Numerical Recipes: The Art of Scientific Computing*", Cambridge University Press (1989 - 2007)
- RM Martin, *"Electronic Structure: basic theory and practical methods",* Cambridge University Press (2004)
- SJ Clark, MD Segall, CJ Pickard, PJ Hasnip, MIJ Probert, K Refson and MC Payne, *"First principles methods using CASTEP",* Zeitschrift für Kristallographie **220**, 567 (2005)