

Electronic optimization

Georg KRESSE

Institut für Materialphysik and Center for Computational Material Science

Universität Wien, Sensengasse 8, A-1090 Wien, Austria



UNIVERSITÄT



WIEN



Overview

- Determination of the electronic groundstate
 - general strategies
 - strategy adopted in VASP
 - iterative matrix diagonalization and mixing
 - how to overcome slow convergence

- molecular dynamics

the algorithms are particularly well suited for molecular dynamics

Density functional theory according to Kohn-Sham

density and kinetic energy:

sum of **one electron** charge densities and kinetic energies

$$\rho^{\text{tot}}(\mathbf{r}) = 2 \sum_{n=1}^{N_e/2} |\psi_n(\mathbf{r})|^2 + \rho^{\text{ion}}(\mathbf{r}), \quad N_e \text{ number of electrons}$$

$$\underbrace{-\frac{\hbar^2}{2m_e} 2 \sum_{n=1}^{N_e/2} \int \psi_n(\mathbf{r})^* \nabla^2 \psi_n(\mathbf{r}) d^3\mathbf{r}}_{\text{kinetic energy}} + \underbrace{\frac{1}{2} \int \frac{\rho^{\text{tot}}(\mathbf{r})\rho^{\text{tot}}(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} d^3\mathbf{r} d^3\mathbf{r}'}_{\text{electrost. energy}} + \underbrace{E_{xc}[\rho(\mathbf{r})]}_{\text{LDA/GGA}}$$

KS-functional has a (the) minimum at the electronic groundstate

Numerical determination of the Kohn-Sham groundstate

- direct minimization of the DFT functional (Car-Parrinello, modern)

start with a set of wavefunctions $\{\psi_n(\mathbf{r}) | n = 1, \dots, N_e/2\}$ (random numbers) and minimizes the value of the functional (**iteration**)

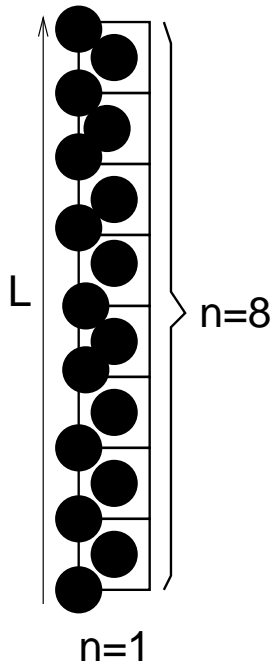
$$\text{Gradient: } F_n(\mathbf{r}) = \left(-\frac{\hbar^2}{2m_e} \nabla^2 + V^{\text{eff}}(\mathbf{r}, \{\psi_n(\mathbf{r}')\}) - \epsilon_n \right) \psi_n(\mathbf{r})$$

- iteration – self consistency (old fashioned)

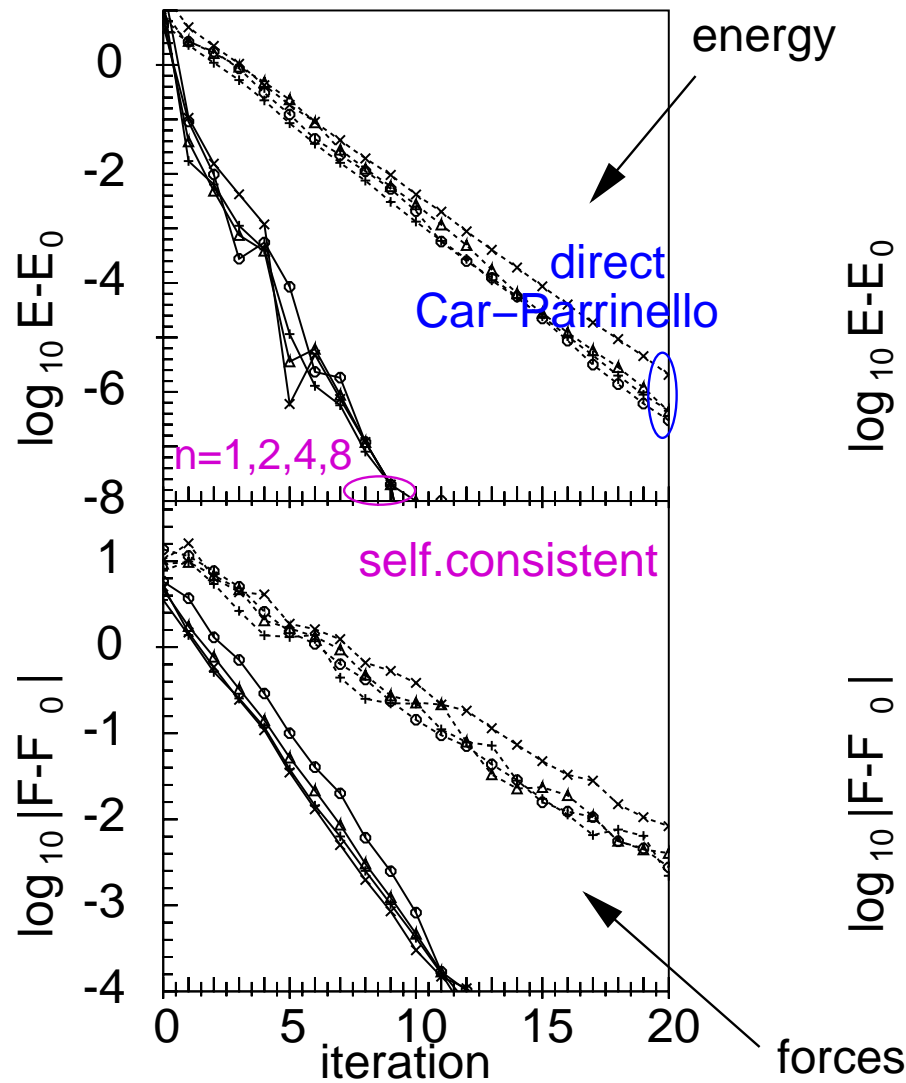
start with a trial density ρ , set up the Schrödinger equation, and solve the Schrödinger equation to obtain wavefunctions $\psi_n(\mathbf{r})$

$$\left(-\frac{\hbar^2}{2m_e} \nabla^2 + V^{\text{eff}}(\mathbf{r}, \{\rho(\mathbf{r}')\}) \right) \psi_n(\mathbf{r}) = \epsilon_n \psi_n(\mathbf{r}) \quad n = 1, \dots, N_e/2$$

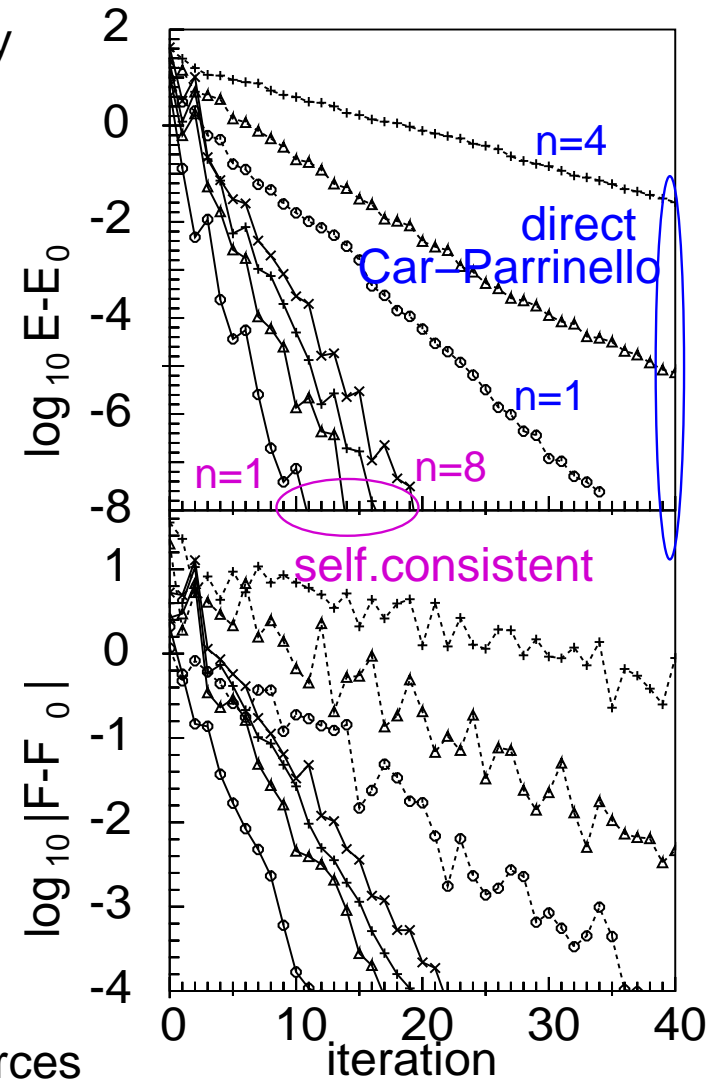
as a result one obtains a new charge density $\rho(\mathbf{r}) = \sum_n |\psi_n(\mathbf{r})|^2$ and a new Schrödinger equation \Rightarrow **iteration**



disordered diamond, insulator



disordered fcc Fe, metal



G. Kresse and J. Furthmüller, Phys. Rev. B **54**, 11169 (1996).

Direct minimization (not supported by vasp.4.5)

- preconditioned conjugate gradient algorithm was applied

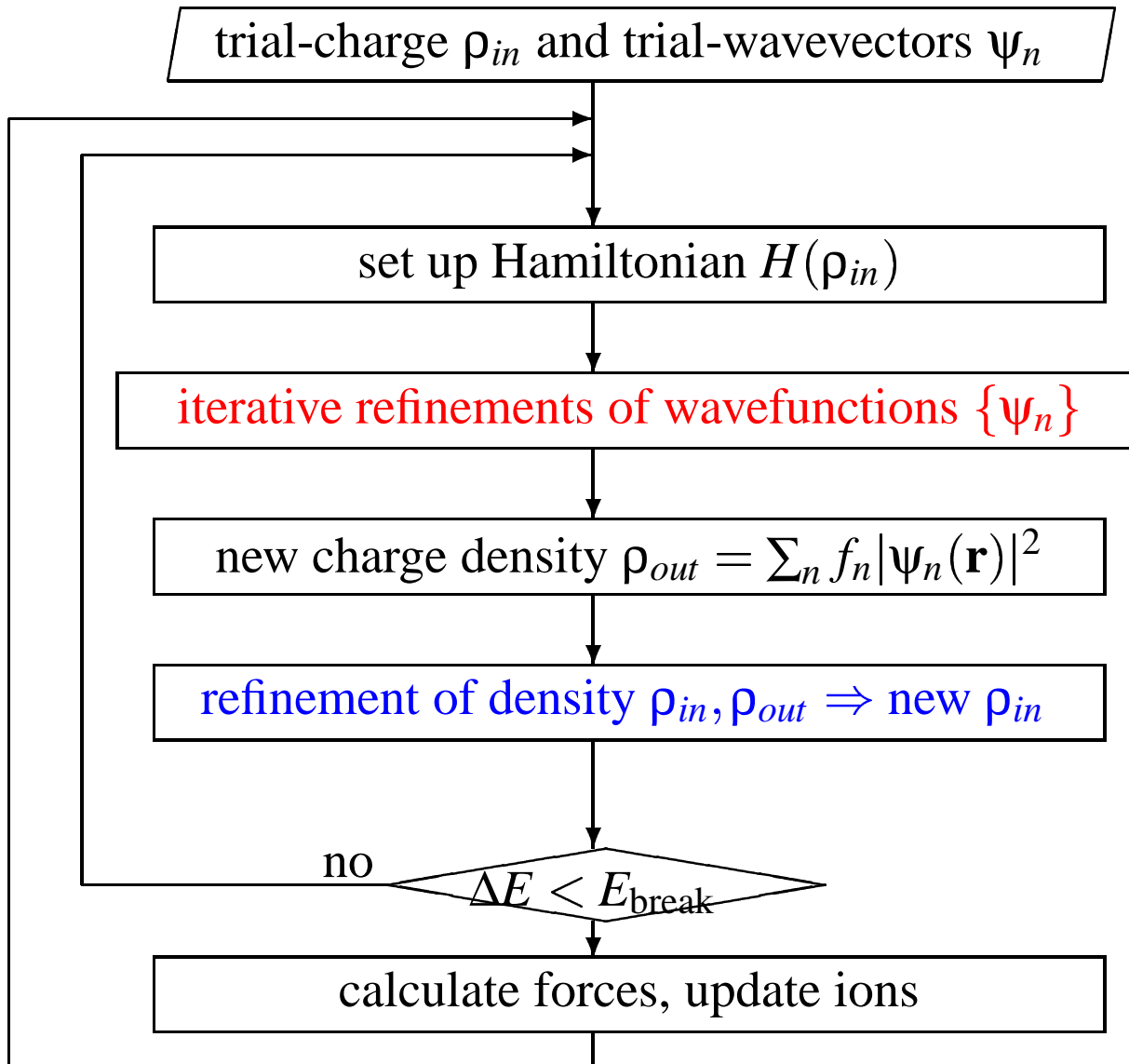
$$\text{Gradient: } F_n(\mathbf{r}) = \left(-\frac{\hbar^2}{2m_e} \nabla^2 + V^{\text{eff}}(\mathbf{r}, \{\psi_n(\mathbf{r}')\}) - \epsilon_n \right) \psi_n(\mathbf{r})$$

- the main troubles are
 - to keep the set of wavefunctions $\{\psi_n(\mathbf{r}) | n = 1, \dots, N_e/2\}$ **orthogonal**
 - “**sub-space**” **rotation**
 - * at the end one aims to have a set of wavefunction $\{\psi_n(\mathbf{r}) | n = 1, \dots, N_{\text{bands}}\}$ that diagonalize the Hamiltonian

$$\langle \psi_n | \mathbf{H} | \psi_m \rangle = \delta_{nm} \bar{\epsilon}_n$$

- * for metals, this condition is difficult to achieve with direct algorithms
- * in metals, actually this optimisation subproblem leads to a linear slowdown with the longest dimension of the (super)cell

Selfconsistency Scheme



- two subproblems
optimization of $\{\psi_n\}$ and ρ_{in}
- refinement of density:
DIIS algorithm
P. Pulay, Chem. Phys. Lett. **73**,
393 (1980).
- refinement of wavefunctions:
DIIS or Davidson algorithm

ALGO flag

- ALGO determines how the wavefunctions are optimized
all algorithms are fully parallel for any data distribution scheme
 - **ALGO= Normal** (default): blocked Davidson algorithm
 - **ALGO= Very Fast**: DIIS algorithm
 - **ALGO= Fast**: 5 initial steps blocked Davidson, afterwards DIIS algorithm
after ions are moved: 1 Davidson step, afterwards again DIIS
- **RMM-DIIS is 1.5 to 2 times faster, but Davidson is more stable**
ALGO= Fast is a very reasonable compromise, and should be specified for system with more than 10-20 atoms
- generally the user can not influence the behavior of these algorithms (delicately optimized black box routines)

OSZICAR *and* OUTCAR files

POSCAR, INCAR and KPOINTS ok, starting setup

WARNING: wrap around errors must be expected

prediction of wavefunctions initialized

entering main loop

	N	E	dE	d eps	ncg	rms	rms (c)
DAV:	1	0.483949E+03	0.48395E+03	-0.27256E+04	96	0.166E+03	
DAV:	2	0.183581E+01	-0.48211E+03	-0.47364E+03	96	0.375E+02	
DAV:	3	-0.340781E+02	-0.35914E+02	-0.35238E+02	96	0.129E+02	
DAV:	4	-0.346106E+02	-0.53249E+00	-0.53100E+00	112	0.158E+01	
DAV:	5	-0.346158E+02	-0.52250E-02	-0.52249E-02	96	0.121E+00	0.198E+01
RMM:	6	-0.286642E+02	0.59517E+01	-0.50136E+01	96	0.584E+01	0.450E+00
RMM:	7	-0.277225E+02	0.94166E+00	-0.47253E+00	96	0.192E+01	0.432E+00

initial charge corresponds to the charge of isolated overlapping atoms (POTCAR)

DAV: blocked Davidson algorithm **RMM:** RMM-DIIS was used

ALGO=F: 5 initial steps blocked Davidson, than RMM-DIIS

4 steps charge fixed, than charge is updated (rms (c) column)

OSZICAR file

N	iteration count
E	total energy
dE	change of total energy
d eps	change of the eigenvalues (fixed potential)
ncg	number of optimisation steps $\mathbf{H}\psi$
rms	total initial residual vector $\sum_{nk} w_k f_{nk} (\mathbf{H} - \epsilon_{nk}) \psi_{nk}$
rms (c)	charge density residual vector

OUTCAR file

initial steps (delay no charge update)

	cpu time	wall clock time	
POTLOK:	VPU time 0.04:	CPU time 0.04	local potential
SETDIJ:	VPU time 0.08:	CPU time 0.08	set PAW strength coefficients
EDDAV :	VPU time 0.94:	CPU time 0.94	blocked Davidson
DOS :	VPU time 0.00:	CPU time 0.00	new density of states

LOOP:	VPU time 1.06:	CPU time 1.06	

charge update:

	cpu time	wall clock time	
POTLOK:	VPU time 0.04:	CPU time 0.04	new local potential
SETDIJ:	VPU time 0.09:	CPU time 0.09	set PAW strength coefficients
EDDIAG:	VPU time 0.14:	CPU time 0.14	sub-space rotation
RMM-DIIS:	VPU time 0.77:	CPU time 0.77	RMM-DIIS step (wavefunc.)
ORTHCH:	VPU time 0.01:	CPU time 0.02	orthogonalisation
DOS :	VPU time -0.01:	CPU time 0.00	new density of states
CHARGE:	VPU time 0.07:	CPU time 0.07	new charge
MIXING:	VPU time 0.01:	CPU time 0.01	mixing of charge

What have all iterative matrix diagonalisation schemes in common ?

- one usually starts with a set of trial vectors (wavefunctions) representing the filled states and a few empty one electron states

$$\{\psi_n | n = 1, \dots, N_{\text{bands}}\}$$

these are initialized using a random number generator

- then the wavefunctions are improved by adding to each a certain amount of the residual vector

the **residual vector** is defined as

$$|R(\psi_n)\rangle = (\mathbf{H} - \epsilon_{\text{app}}\mathbf{S})|\psi_n\rangle \quad \epsilon_{\text{app}} = \langle \psi_n | \mathbf{H} | \psi_n \rangle$$

- adding a small amount of the residual vector

$$\psi_n \rightarrow \psi_n + \lambda R(\psi_n)$$

is in the spirit of the steepest descent approach (usually termed “Jacobi relaxation”)

Iterative matrix diagonalization based on the DIIS algorithm

- for our case we need a rather **specialized eigenvalue solver**
 - it should be capable of doing only **little work**
 - efficiency and **parallelization** are important issues
- **two step procedure**
 - start with a set of trial vectors (wavefunctions) representing the electrons $\{\psi_n | n = 1, \dots, N_{\text{bands}}\}$ (initialized with random numbers)
 - apply Rayleigh Ritz optimization in the space spanned by all bands (“**sub-space rotation**”)
transform: $\{\psi_n | n = 1, \dots, N_{\text{bands}}\}$ so that
$$\langle \psi_n | \mathbf{H} | \psi_m \rangle = \delta_{nm} \bar{\epsilon}_n$$
 - then **optimize each vector individually** $\{\psi_n | n = 1, \dots, N_{\text{bands}}\}$ two or three times

“In space” optimization EDDIAG

- a set of vectors, that represent the valence electrons $\{\psi_n | n = 1, \dots, N_{\text{bands}}\}$
- Raghly Ritz optimization in the space spanned by these vectors (subspace) search for the unitary matrix \bar{U} such that the wavefunctions ψ'_n

$$\psi'_n = \sum_m \bar{U}_{mn} \psi_m$$

fulfill

$$\langle \psi'_n | \mathbf{H} | \psi'_m \rangle = \epsilon_m \delta_{nm}$$

this requires the calculation of the subspace matrix \bar{H}

$$\langle \psi_n | \mathbf{H} | \psi_m \rangle = \bar{H}_{mn} \quad (\langle \psi_n | \mathbf{S} | \psi_m \rangle = \delta_{mn} \quad \text{always holds})$$

and it's diagonalisation

- the setup of the matrix scales like $N_{\text{bands}}^2 N_{\text{FFT}}$ (worst scaling part of VASP) in the parallel version, communication is required, but modest worse is the fact that the diagonalisation of \bar{H}_{mn} is not done in parallel

Iterative matrix diagonalization based on the DIIS algorithm

- “out of space optimization” EDDRMM
 - minimize norm of residual vector using the **DIIS method**

$$|R(\psi_n)\rangle = (\mathbf{H} - \epsilon_{\text{app}}\mathbf{S})|\psi_n\rangle \quad \langle R(\psi_n)|R(\psi_n)\rangle \rightarrow \text{minimal}$$

- each vector is optimized **individually** (without regard to any other vector)
- easy to implement on parallel computers since each processor handles a subset of the vectors (no communication required, NPAR=number of proc.)
- scaling is proportional to $N_{\text{bands}}N_{\text{FFT}}$ with a large prefactor dominates the compute time for medium to large problems
- orthogonalization of wavefunctions ORTHCH

Problem of the DIIS algorithm

- eigenstates can be missed for large systems
and there is no clear way to tell when this happens
 - in the “best” case no convergence
 - but convergence might also slows down after reaching a precision of 10^{-2} or 10^{-3}
 - in the worst case, you might not notice anything
- in these cases, switch to blocked Davidson (manual contains a number of tricks how you might be able to use the DIIS algorithm even when it initially fails)
- things are not that bad
if the Davidson algorithm is used for the first steps, there is practically no danger of missing eigenstates

VASP.4.5: new blocked Davidson algorithm

- combines “in space” and “out of space” optimization
- selects a subset of all bands $\{\psi_n | n = 1, \dots, N_{\text{bands}}\} \Rightarrow \{\psi_k | k = n_1, \dots, n_2\}$
 - optimize this subset by adding the orthogonalized residual vector to the presently considered subspace
$$\{\psi_k, (\mathbf{H} - \epsilon_{\text{app}}\mathbf{S})\psi_k | k = n_1, \dots, n_2\}$$
 - apply Rayleigh Ritz optimization in the space spanned by these vectors (“sub-space” rotation in a $2(n_2 - n_1 + 1)$ dim. space)
 - add additional residuals calculated from the yet optimized bands (“sub-space” rotation in a $3(n_2 - n_1 + 1)$ dim. space)
- approximately a factor of 1.5-2 slower than RMM-DIIS, but always stable
- available in **parallel** for any data distribution

charge density mixing (RMM-DIIS)

- VASP aims at the minimization of the norm of residual vector

$$R[\rho_{\text{in}}] = \rho_{\text{out}}[\rho_{\text{in}}] - \rho_{\text{in}} \quad |R[\rho_{\text{in}}]| \Rightarrow \min$$

with $\rho_{\text{out}}(\vec{r}) = \sum_{\text{occupied}} w_k f_{nk} \psi_{nk}(\vec{r})^2$

- **DIIS algorithm** is used for the optimization of the norm of the residual vector
- linearization of $R[\rho_{\text{in}}]$ around ρ_{sc} (linear response theory)

$$R[\rho] = -\mathbf{J}(\rho - \rho_{\text{sc}}),$$

with the charge dielectric function \mathbf{J}

$$\mathbf{J} = \mathbf{1} - \chi \underbrace{\mathbf{U}}_{\frac{4\pi e^2}{q^2}},$$

leads to

$$R[\rho_{\text{in}}] = \rho_{\text{out}}[\rho_{\text{in}}] - \rho_{\text{in}} = \mathbf{J}(\rho_{\text{in}} - \rho_{\text{sc}})$$

Divergence of the dielectric function

eigenvalue spectrum of **J** determines convergence

$$\mathbf{J} = \mathbf{1} - \chi \underbrace{\mathbf{U}}_{\frac{4\pi e^2}{q^2}},$$

“broader” eigenvalue spectrum \Rightarrow slower convergence

- for insulators and semi-conductors the width of the eigenvalue spectrum is constant and system size independent !
- for metals the eigenvalue spectrum **diverges**, its width is proportional to the square of the longest dimension of the cell:
 - short wavelength limit $\mathbf{J} \approx \mathbf{1}$ (no screening)
 - long wavelength limit $\mathbf{J} \approx \mathbf{1}/\mathbf{q}^2 \propto \mathbf{L}^2$ (metallic screening)

complete screening in metals causes slow convergence to the groundstate (charge sloshing)

VASP charge density mixer

- VASP uses a model dielectric function which is a good initial approximation for most systems

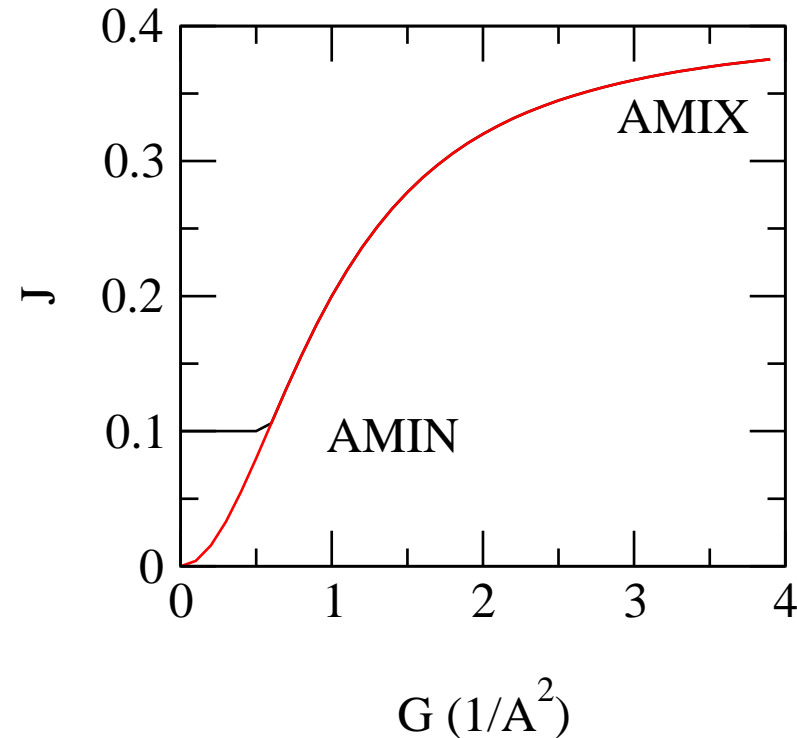
$$\mathbf{J}^{-1} \approx \mathbf{G}_q^1 = \max\left(\frac{q^2 \text{AMIX}}{q^2 + \text{BMIX}}, \text{AMIN}\right)$$

- defaults:

AMIX=0.4 ; AMIN=0.1 ;

BMIX=1.0

- this is combined with a **convergence accelerator**
the initial guess for the dielectric matrix is improved using information accumulated in each electronic (mixing) step
direct inversion in the iterative subspace (DIIS)



How can you tune VASP to achieve faster convergence

- try **linear mixing** (AMIX=0.1-0.2, BMIX=0.0001)

$$\mathbf{J}^{-1} \approx \mathbf{G}_q^1 = A$$

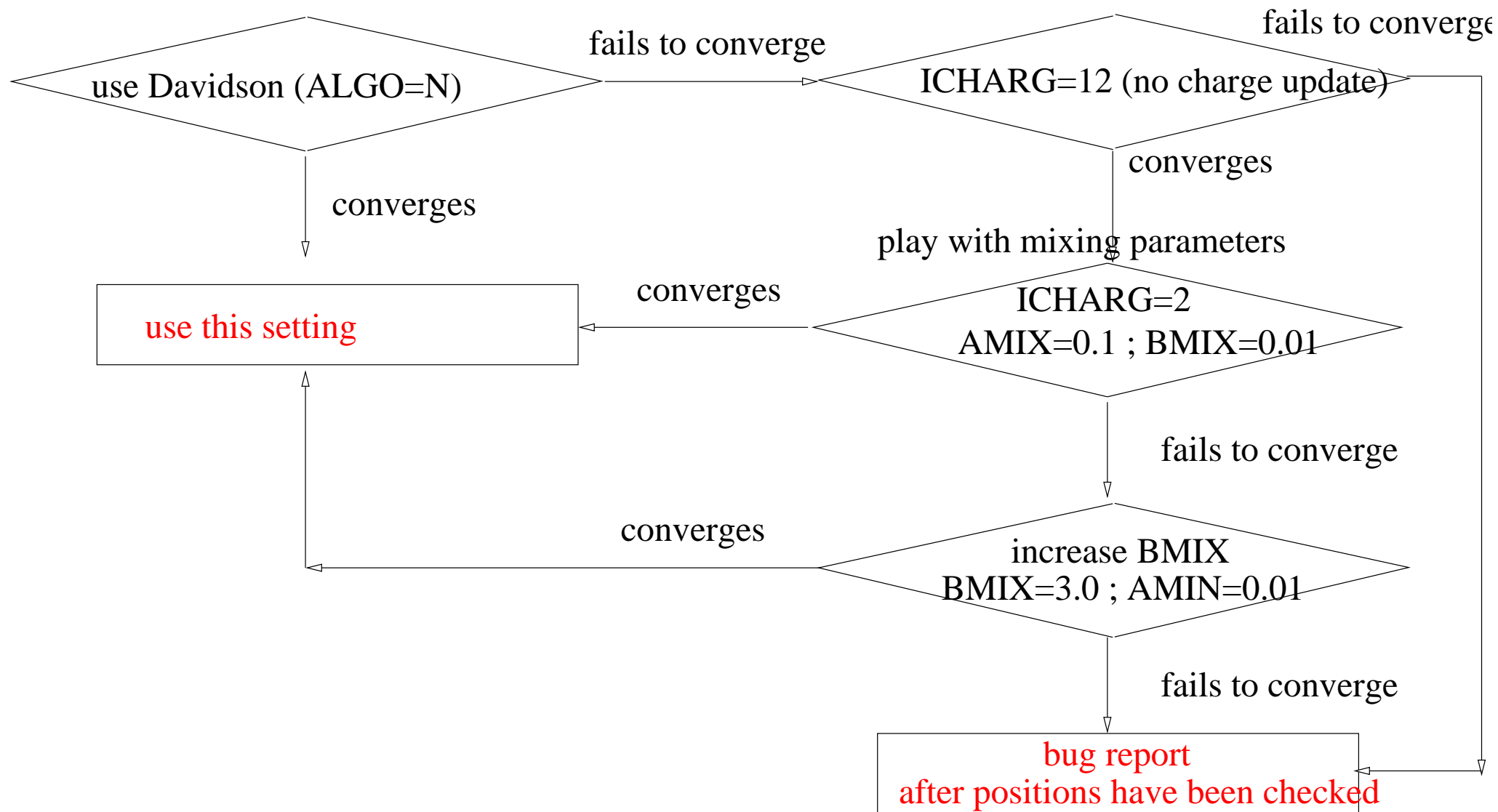
- VASP also gives information on how good the initial mixing parameters are
allow VASP to run until selfconsistency is achieved and search for the last occurrence of

eigenvalues of (default mixing * dielectric matrix)

average eigenvalue GAMMA= 2.200

- for linear mixing (e.g. AMIX=0.1 ; BMIX=0.0001) the optimal AMIX is given by the present AMIX×GAMMA
- Kerker like mixing (model dielectric matrix):
 - * GAMMA larger 1 → decrease BMIX
 - * GAMMA smaller 1 → increase BMIX

What to do when electronic convergence fails



ab initio Molecular dynamics

CP approach

elegant
simple to implement

problematic for metals, since
**electrons must decouple from ionic
degrees of freedom**
not the case for metals

small timestep

exact KS-groundstate

large timestep

direct minimization

problematic for metals
large memory requirements
damped second order
(Tassone, Mauri, Car)
conjugate gradient
(Arias, Payne, Joannopoulos)
RMM-DIIS
(Hutter, Lüthi, Parrinello)

selfconsistency cycle

very stable
efficient for insulators
and metals

Selfconsistency cycle is very well suited for MDs

- MD on the Born Oppenheimer surface (exact KS-groundstate)
- **selfconsistency cycle determines the dielectric matrix**
first time step is rather expensive
but since the dielectric matrix changes only little when ions are moved, the method becomes very fast in successive steps
- wavefunctions and charges etc. are “forward” extrapolated between time-steps
- all this makes an extremely efficient scheme that is competitive with the so called “Car-Parrinello” scheme for insulators
for metals, our scheme is generally much more robust and efficient than the Car-Parrinello scheme
- to select this feature in VASP, set MAXMIX in the INCAR file

Using MAXMIX

- usually VASP **resets the dielectric matrix to its default after moving the ions**
but if the ions move only a little bit one can bypass this reset
 - definitely a good option for molecular dynamics
 - damped molecular dynamics (optimisation)
 - works also well during relaxations, if the forces are not large (<0.5 eV/Å)
- you need to specify MAXMIX in the INCAR file
set MAXMIX to roughly three times the number of iterations in the first ionic step
the resulting speedups can be substantial (a factor 2 to 3 less electronic steps for each ionic step)

Using Molecular dynamics

a simple INCAR file

```
ENMAX = 250 ; LREAL = A      # electronic degrees
ALGO = V                    # very fast algorithm
MAXMIX = 80                 # mixing

IBRION = 0                  # MD
NSW = 1000                  # number of MD steps
POTIM = 3.0                 # time step
TEBEG = 1500 ; TEEND = 500 # target temperature 1500-500 K
SMASS = -1 ; NBLOCK = 50   # scale velocities every 50 steps
SMASS = 2                   # use a Nose Hoover thermostat
SMASS = -3                  # micro canonical
```

Using Molecular dynamics

- timestep POTIM, depends on the vibrational frequencies and the required energy conservation

as a rule of thumb: increase POTIM until 3 electronic minimisation steps are required per timestep

another rule of thumb:

H	0.5 fs	increase by 1 fs for each row
Li-F	1 fs	

- SMASS controls the MD simulation
 - SMASS=-3 micro canonical ensemble
 - for equilibration and simulated annealing SMASS = -1 ; NBLOCK = 50-100 microcanonical MD, and every NBLOCK steps the kinetic energy is scaled to meet the required temperature criterion
 - for positive values a Nose Hoover thermostat is introduced