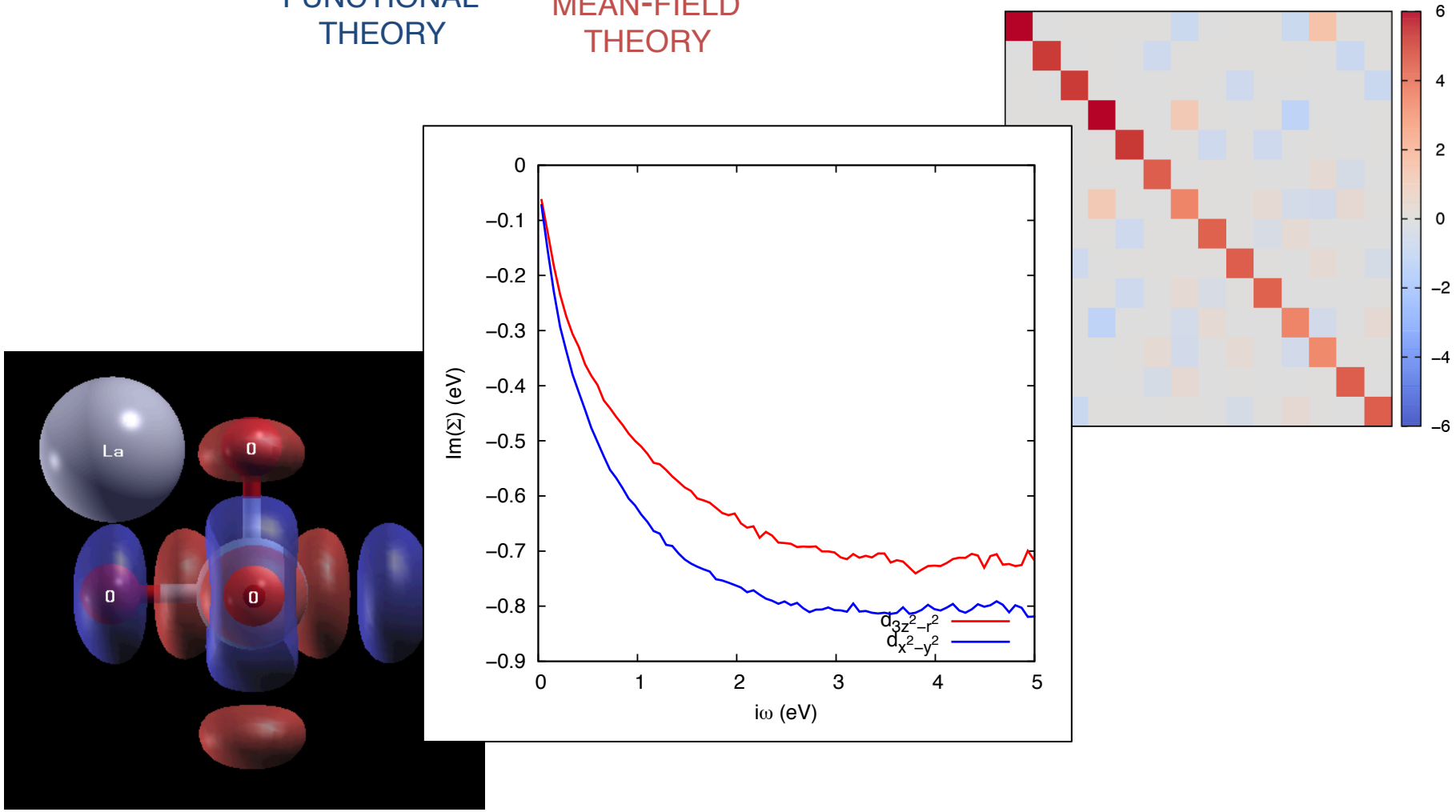
# DFT + DMFT in Practice

DENSITY FUNCTIONAL THEORY

DYNAMICAL MEAN-FIELD THEORY




**Eric Isaacs**

Department of Applied Physics and Applied Math, Columbia University

# Caveats

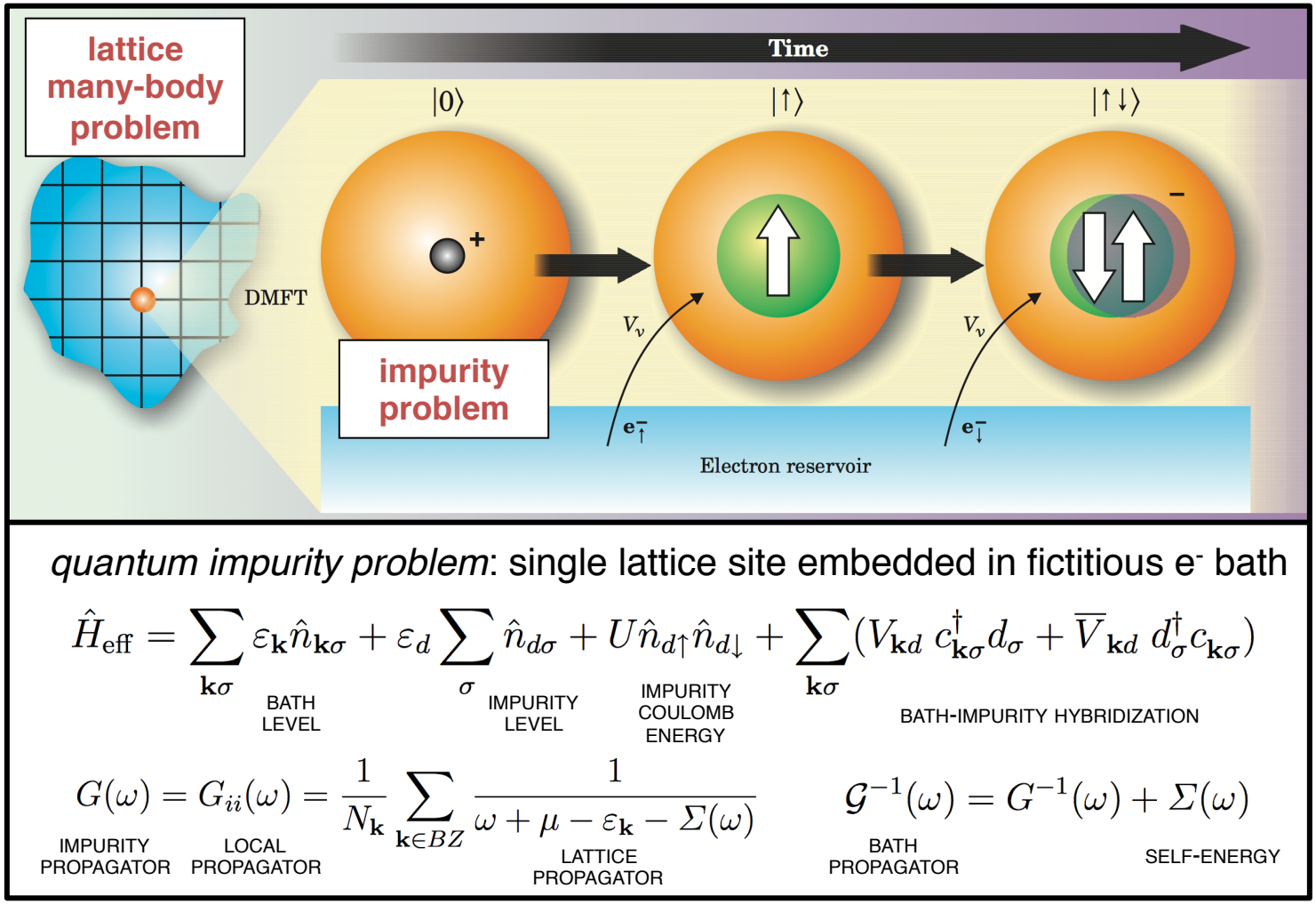
1) Will not be getting into the theory [see later tutorials]

2) Will only be discussing **single-site** DMFT, **no charge self-consistency**

3) Will only be discussing one particular framework (**VASP**, **WANNIER90**, **Haule's CTQMC**, **Hyowon's DMFT.PY**)

4) I'm still learning this so please (especially Chris, Jia, and Hanghui) chime in to correct/clarify

# Motivation: DMFT[1-3] for realistic electronic structure calculations




*quantum impurity problem*: single lattice site embedded in fictitious e⁻ bath

$$\hat{H}_{\text{eff}} = \sum_{\mathbf{k}\sigma} \varepsilon_{\mathbf{k}} \hat{n}_{\mathbf{k}\sigma} + \varepsilon_d \sum_{\sigma} \hat{n}_{d\sigma} + U \hat{n}_{d\uparrow} \hat{n}_{d\downarrow} + \sum_{\mathbf{k}\sigma} (V_{\mathbf{k}d}\, c^\dagger_{\mathbf{k}\sigma} d_\sigma + \overline{V}_{\mathbf{k}d}\, d^\dagger_\sigma c_{\mathbf{k}\sigma})$$

BATH LEVEL · IMPURITY LEVEL · IMPURITY COULOMB ENERGY · BATH-IMPURITY HYBRIDIZATION

$$G(\omega) = G_{ii}(\omega) = \frac{1}{N_{\mathbf{k}}} \sum_{\mathbf{k}\in BZ} \frac{1}{\omega + \mu - \varepsilon_{\mathbf{k}} - \Sigma(\omega)}$$

IMPURITY PROPAGATOR · LOCAL PROPAGATOR · LATTICE PROPAGATOR

$$\mathcal{G}^{-1}(\omega) = G^{-1}(\omega) + \Sigma(\omega)$$

BATH PROPAGATOR · SELF-ENERGY

**exact in the limit of infinite dimension** or lattice coordination

*many-body* description of the **dynamical local correlations** (only static in DFT+U)

strategy: employ **DMFT** for correlated states and **DFT** for the rest[4]

[1] W. Metzner and D. Vollhardt, PRL **62**, 324 (1989). [2] A. Georges and G. Kotliar, PRB **45**, 6479 (1992). [3] G. Kotliar and D. Vollhardt, Phys. Today **57**, 53 (2004). [4] G. Kotliar et al., Rev. Mod. Phys. **78**, 865 (2006).

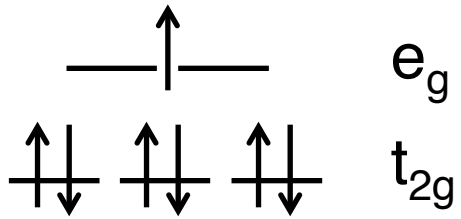
# Terminology

| | |
|---|---|
| **Impurity** | single lattice site on which we treat the correlations |
| **Bath** | auxiliary electron reservoir mimicking the effect of the discarded lattice sites |
| **Hybridization** | coupling b/w impurity and bath, as described by $\Delta(\omega) = V^2/[\omega-(\varepsilon-\mu)]$ summed over bath states |
| **Impurity Green function** $G(\omega)$ | Green function of the impurity problem |
| **Bath Green function** $\mathcal{G}(\omega)$ | non-interacting Green function of the impurity problem |
| **Lattice Green function** $G(k,\omega)$ | Green function of the lattice at a particular *k*-point |
| **Local Green function** $G_{ii}(\omega)$ | Lattice Green function summed over *k*-points |
| **Self-energy** $\Sigma(\omega)$ | interaction-induced shift (real part) and broadening (imaginary part) of 1-particle energy levels |

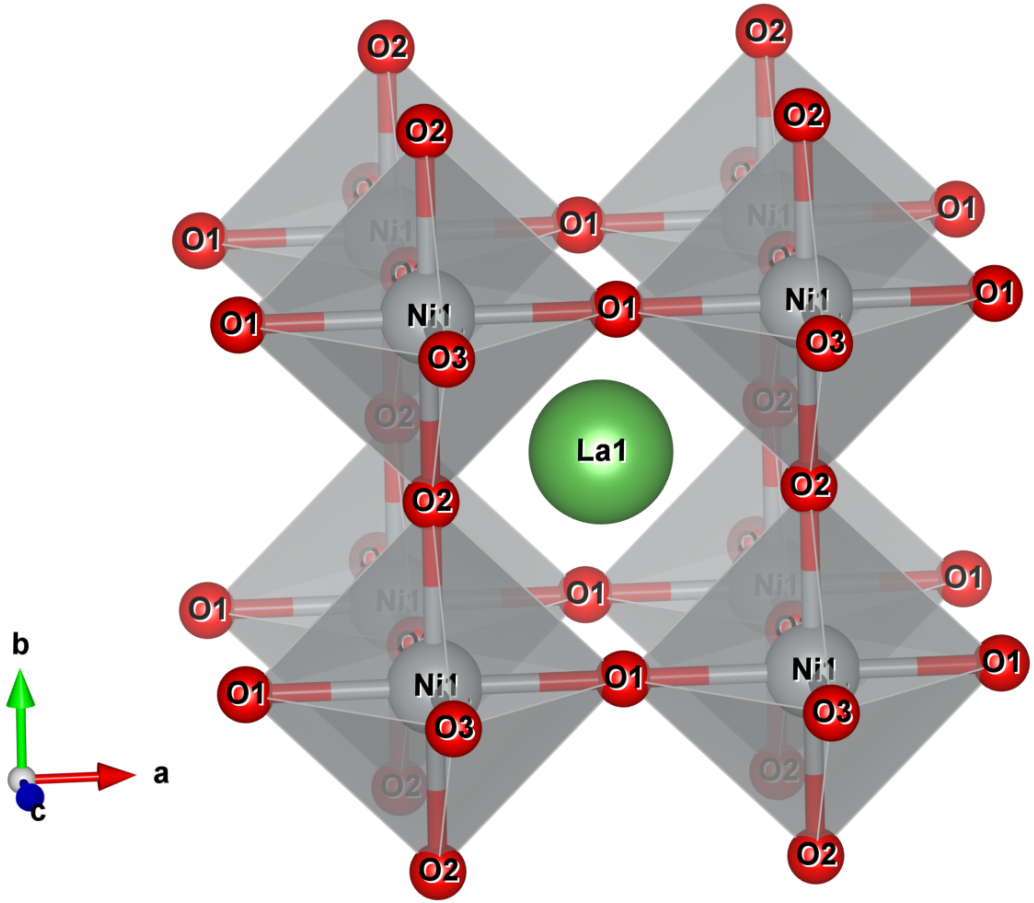
# Example: $LaNiO_3$

| La | Ni | $O_3$ |
|---|---|---|
| 3+ | 3+ | 6- |

Ni has 10 valence electrons so $Ni^{3+}$ has 7 *d* electons



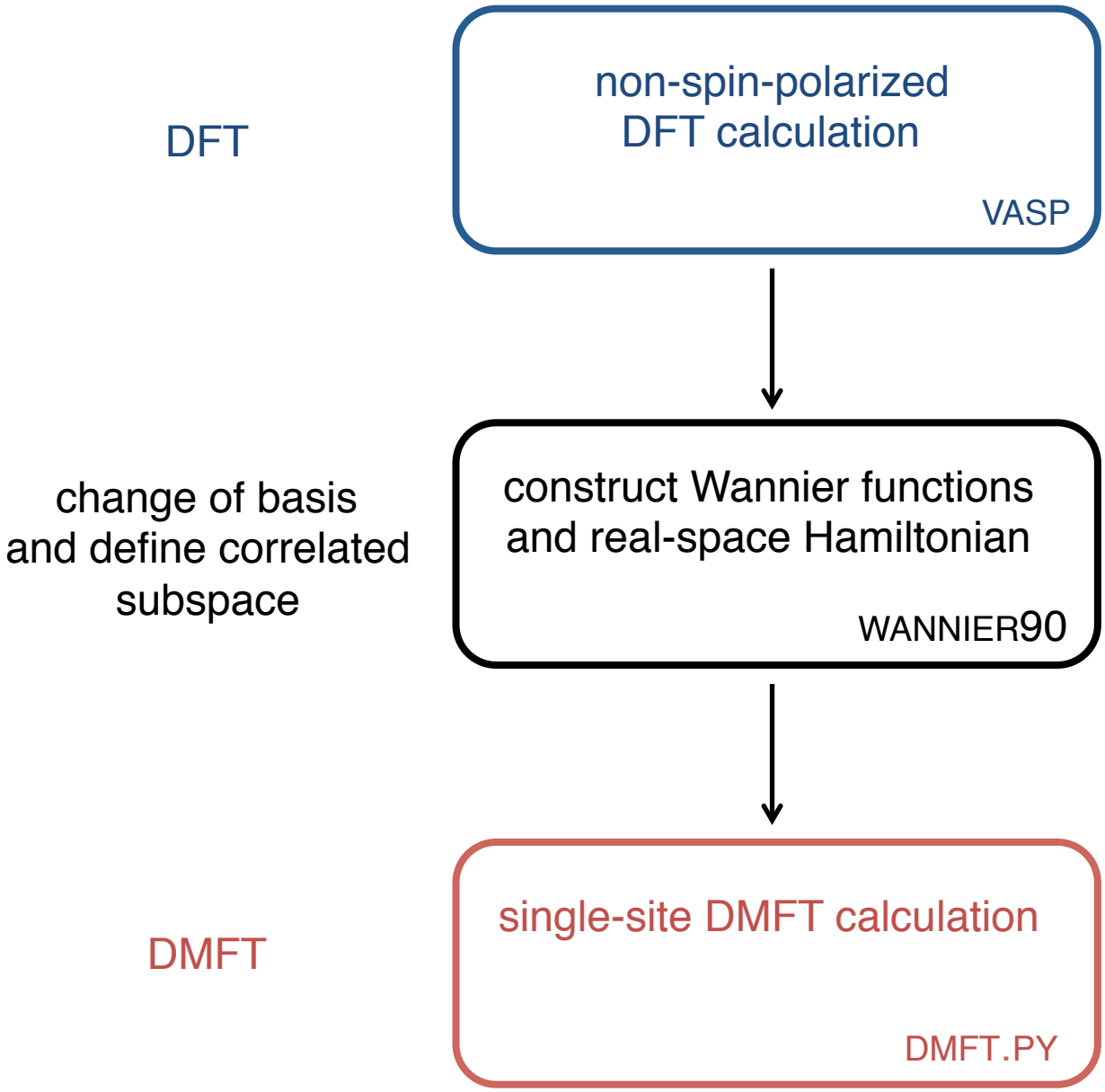

$t_{2g}$ filled, 1 $e^-$ in $e_g$




slight tetragonal distortion (c/a = 0.963)

*will use this example as we proceed…*

# Basic Workflow

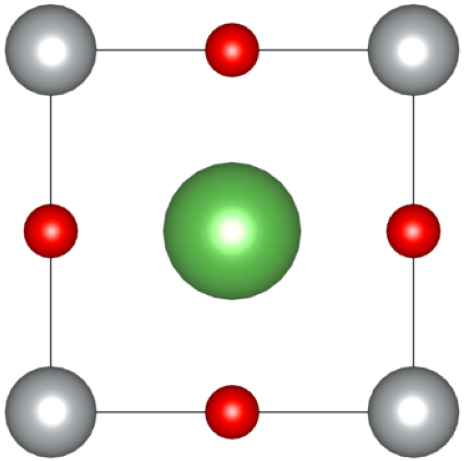


# Part 1: DFT

input

INCAR file

NBANDS = 24
ISYM = 2
**ISPIN = 1** ← non-spin-polarized
LDAU = .TRUE.
LDAUTYPE = 1
LDAUL = -1 2 -1
**LDAUU = 0 0.0 0**
**LDAUJ = 0 0.0 0** } DFT (U=J=0)
LDAUPRINT = 1
LMAXMIX = 6
LASPH = .TRUE.
**LORBIT = 11** ← so we get the PDOS
EDIFF = 1E-6
NELM = 50
ISMEAR = -5
ENCUT = 600

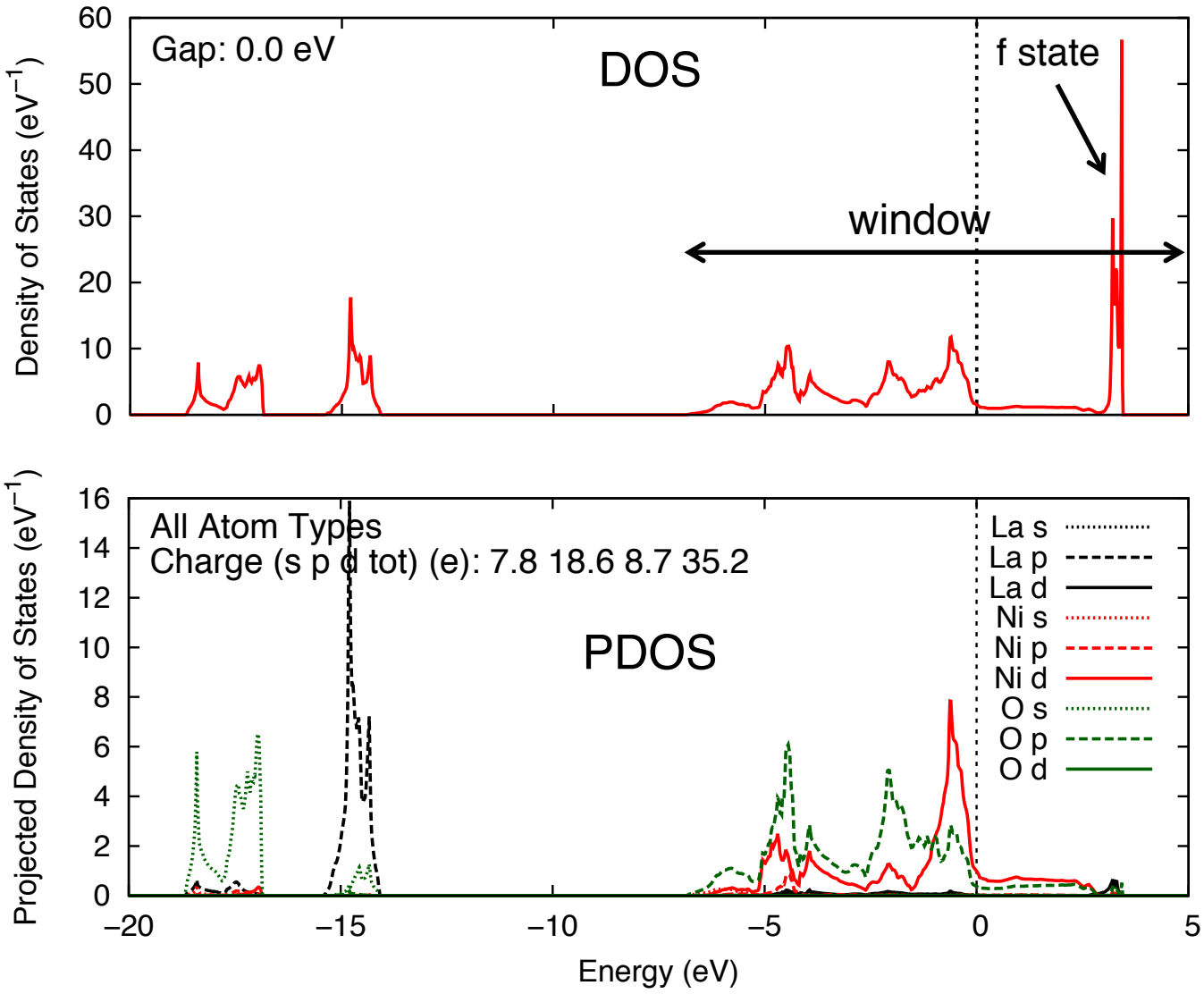


see http://grandcentral.apam.columbia.edu:5555/tutorials/
for general VASP tutorial

VASP

# Part 1: DFT

output




VASP

# Part 1: DFT

input

INCAR file

NBANDS = 24
ISYM = 2
ISPIN = 1
LDAU = .TRUE.
LDAUTYPE = 1
LDAUL = -1 2 -1
LDAUU = 0 0.0 0
LDAUJ = 0 0.0 0
LDAUPRINT = 1
LMAXMIX = 6
LASPH = .TRUE.
LORBIT = 11
EDIFF = 1E-6
NELM = 50
ISMEAR = -5
ENCUT = 600
**LWANNIER90 = .TRUE.**
**LWRITE_UNK = .TRUE.**

← flags to generate WANNIER90 input

5 Ni *d* states
+ 3 × 3 O *p* states →

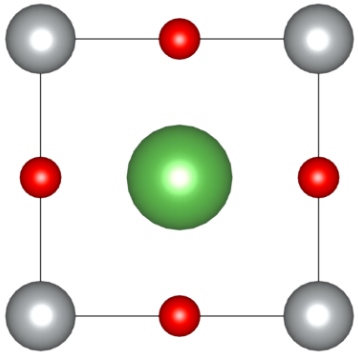
wanner90.win file

dis_win_min = -1
dis_win_max = 11
num_wann = 14

this is the window including all the Ni *d* states

WARNING: These refer to energies not shifted by the Fermi energy

begin projections
Ni:d
O:p
end projections



VASP compiled with WANNIER90 flags

# Part 2: Wannier

input

$M_{mn}^{(\mathbf{k},\mathbf{b})} = \langle u_{m\mathbf{k}} | u_{n\mathbf{k}+\mathbf{b}} \rangle$ overlaps (.mmn files)

$A_{mn}^{(\mathbf{k})} = \langle \psi_{m\mathbf{k}} | g_n \rangle$ projections (.amn files)

generated by VASP

VASP will update wannier90.win w/ pertinent input data (e.g. structure and k-points); you should also add additional flags:

wanner90.win file

```
wannier_plot = .true.
wannier_plot_format = xcrysden
wannier_plot_supercell = 2 2 2
wannier_plot_list = 1-14

num_iter = 1000
dis_num_iter = 500

begin kpoint_path
R 0.5 0.5 0.5  G 0.0 0.0 0.0
G 0.0 0.0 0.0  X 0.0 0.5 0.0
X 0.0 0.5 0.0  M 0.5 0.5 0.0
M 0.5 0.5 0.0  G 0.0 0.0 0.0
end kpoint_path

bands_plot = .true.
bands_plot_format = gnuplot
bands_num_points = 30
```

see www.wannier.org for complete documentation

WANNIER90 w/ modified plot.F90

# Part 2: Wannier

output

$$w_{n\mathbf{R}}(\mathbf{r}) = \frac{V}{(2\pi)^3} \int_{\mathrm{BZ}} \left[ \sum_m U_{mn}^{(\mathbf{k})} \psi_{m\mathbf{k}}(\mathbf{r}) \right] e^{-\mathrm{i}\mathbf{k}.\mathbf{R}} \, \mathrm{d}\mathbf{k}$$

Wannier function w/ *U* chosen to minimize spread $\langle r^2 \rangle - \langle r \rangle^2$

$$\langle w_{n\mathbf{0}} | H_{KS} | w_{m\mathbf{R}} \rangle$$

real-space Hamiltonian (rham.py)

spread in Å$^2$

```
$ grep CONV wannier90.wout | tail -n 5
   996    -0.917E-12     0.0000027420     12.4962776151   2677.47  <-- CONV
   997    -0.128E-12     0.0000027396     12.4962776151   2677.66  <-- CONV
   998    -0.167E-12     0.0000027393     12.4962776151   2677.86  <-- CONV
   999    -0.194E-12     0.0000027388     12.4962776151   2678.05  <-- CONV
  1000     -0.302E-13     0.0000027383     12.4962776151   2678.25  <-- CONV
```

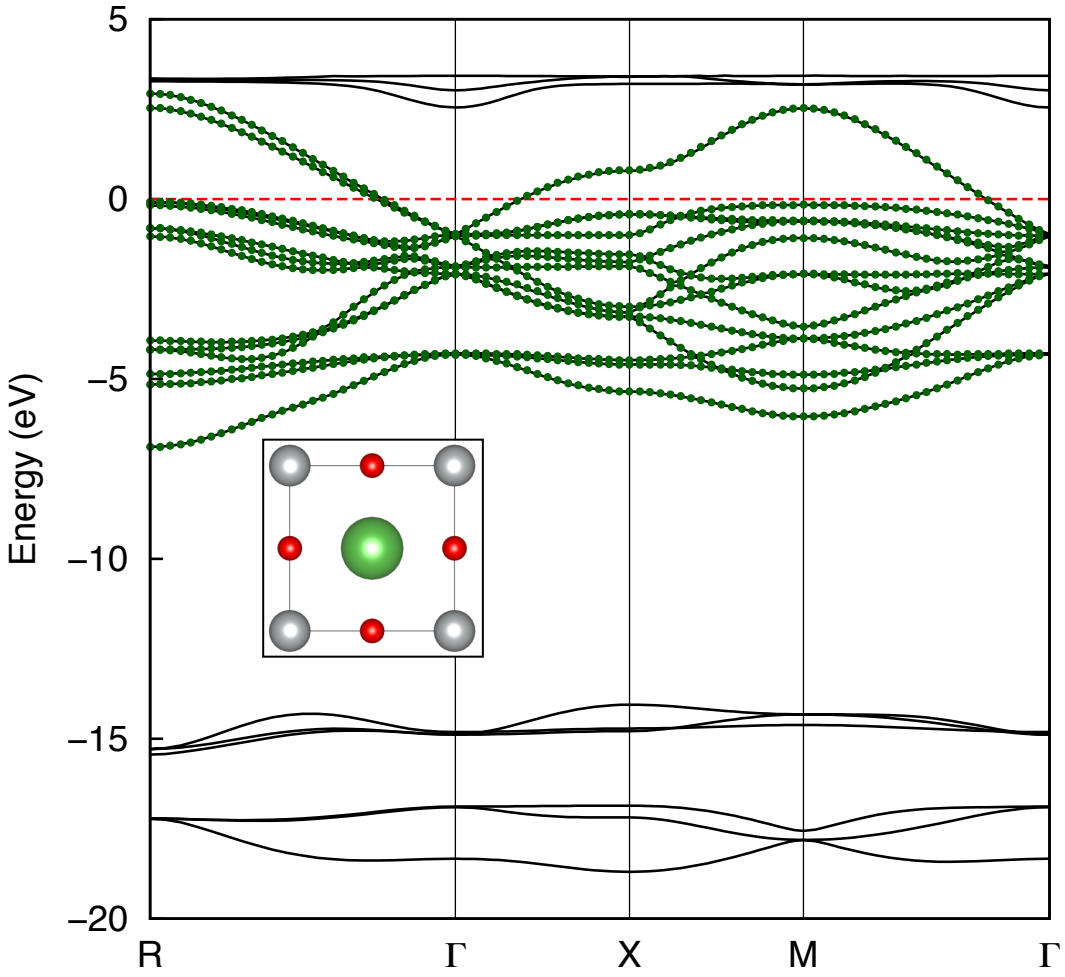
iteration

output log (wannier90.out)

WANNIER90 w/ modified plot.F90

# Part 2: Wannier

output




DFT bands successfully reproduced

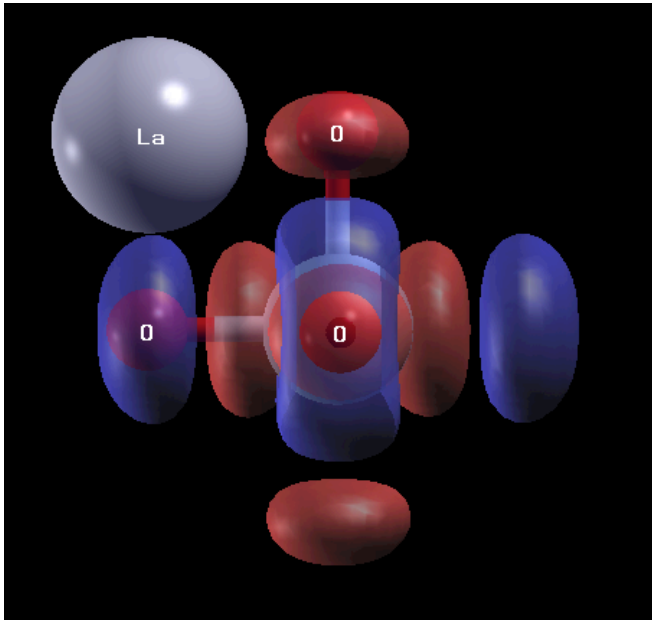
DFT

Wannier

WANNIER90 w/ modified plot.F90

# Part 2: Wannier

output



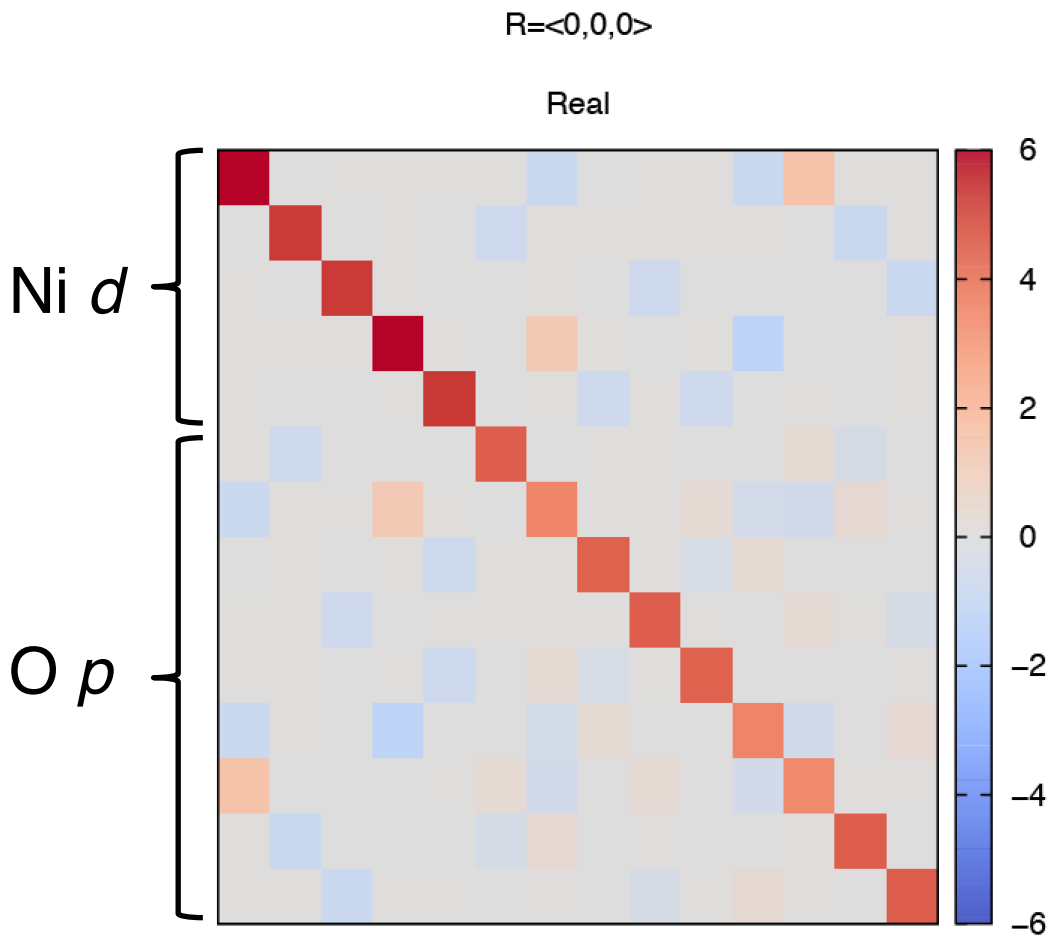

we get Wannier functions centered on the Ni and O sites

e.g. Ni $3z^2$-$r^2$-like d orbital

WANNIER90 w/ modified plot.F90

# Part 2: Wannier

output



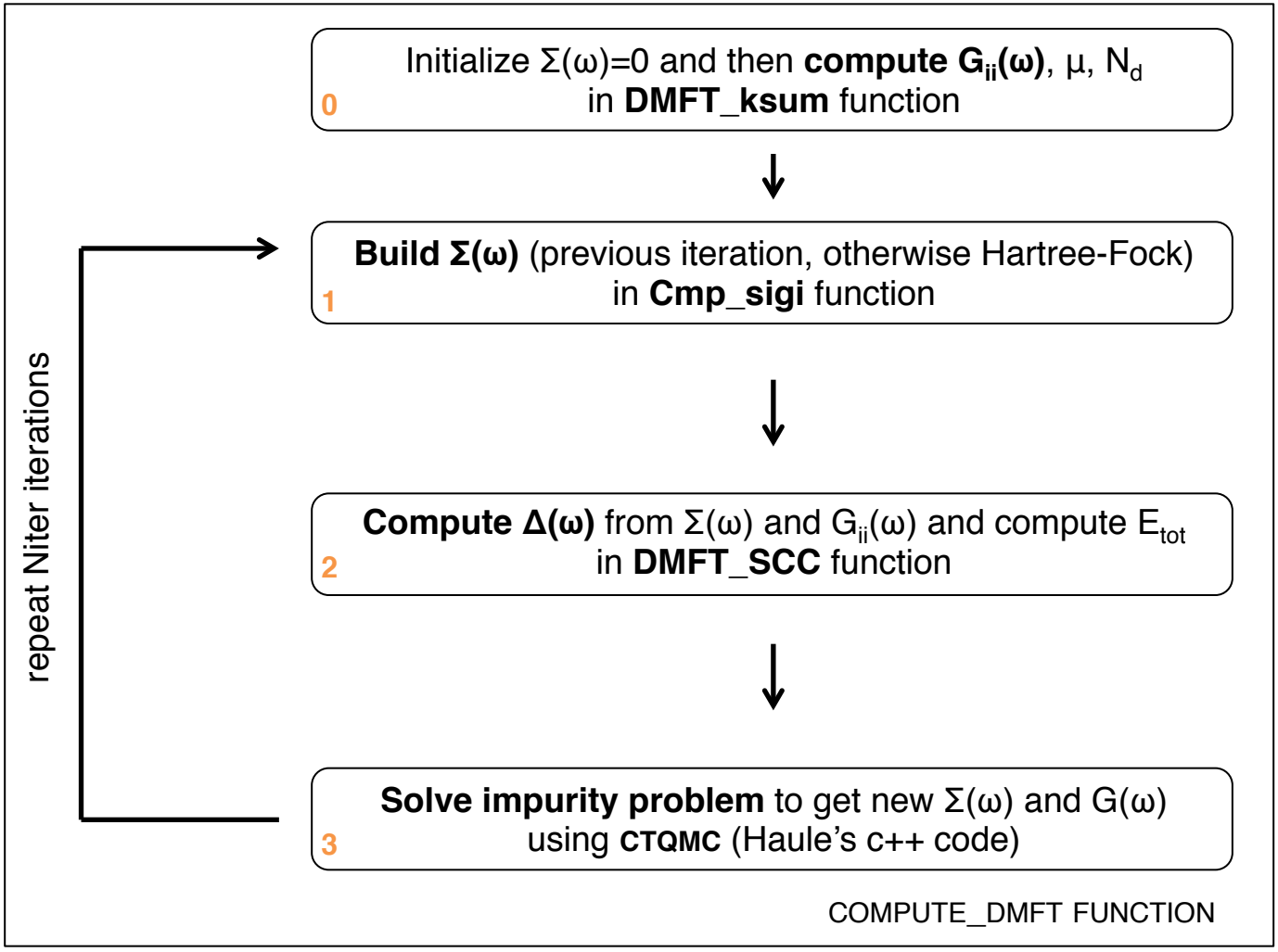

real-space Hamiltonian (rham.py)

Note: can be useful to do local coordinate transformation to minimize off-diagonal elements in Ni d block [see Appendix A of PRB **90**, 235103 (2014)]

WANNIER90 w/ modified plot.F90

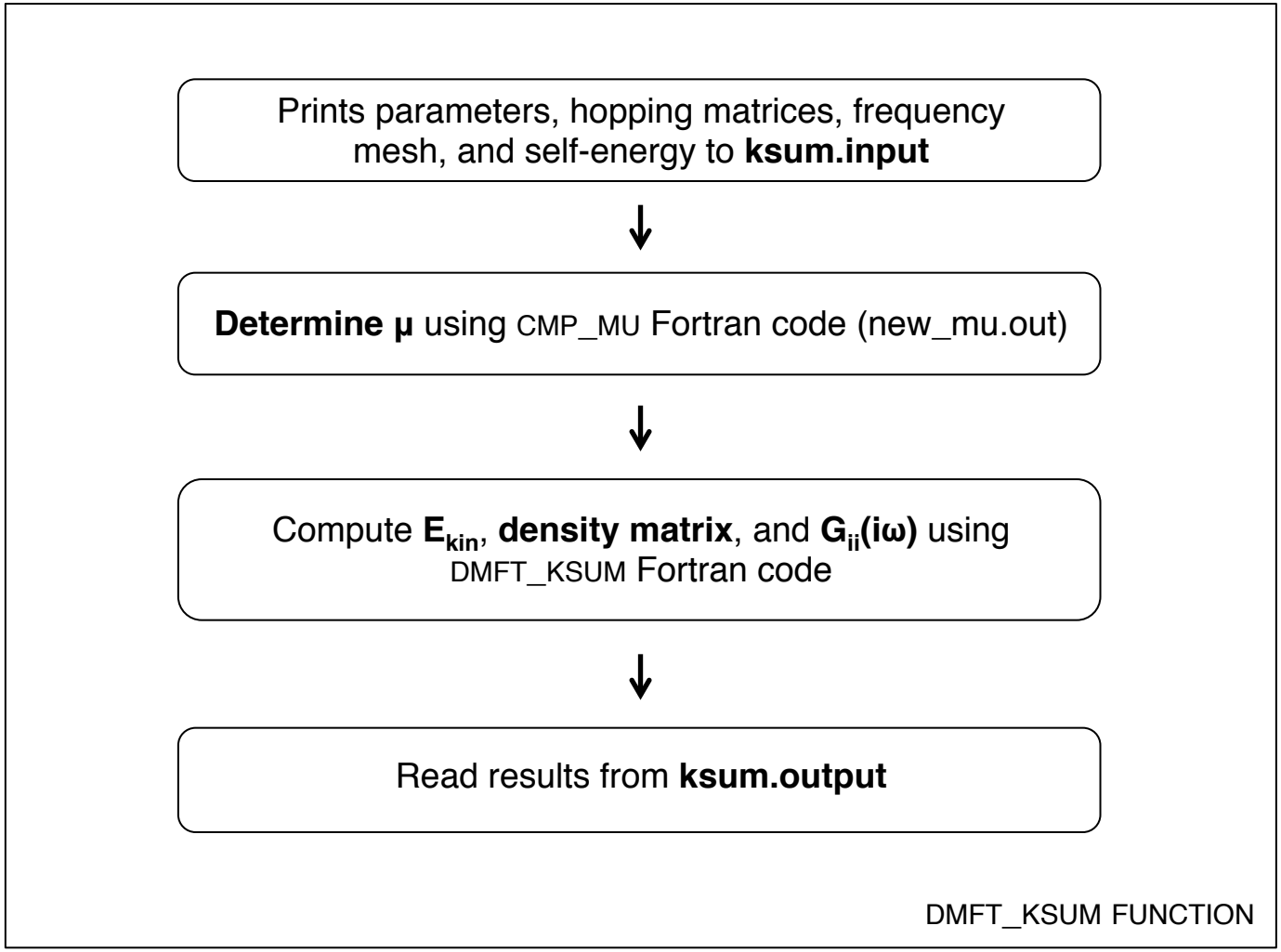
# DMFT Workflow

RUN_DMFT.PY reads in Hopping, params, params_ctqmc and calls DMFT.Compute_DMFT




DMFT.PY

# DMFT: dmft_ksum




DMFT.PY

# DMFT: other components

FILEIO.PY for reading and writing data

DMFT.LINEAR_INTERPOLATE and SCIPY.INTERPOLATE for interpolation

STRUCT.PY for reading DMFTPOSCAR

DMFT.CREATEINPUTFILE to generate PARAMS for CTQMC

GENERATE_CIX.PY for generating CTQMC impurity input data

DMFT.PY

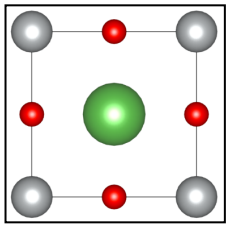
# Part 3: DMFT

input

```
LaNiO3
1.0
3.94774481 0.00000000 0.00000000
0.00000000 3.94774481 0.00000000
0.00000000 0.00000000 3.80244673
5 9
Direct
0.00000000 0.00000000 0.00000000 d_z2
0.00000000 0.00000000 0.00000000 d_xz
0.00000000 0.00000000 0.00000000 d_yz
0.00000000 0.00000000 0.00000000 d_x2y2
0.00000000 0.00000000 0.00000000 d_xy
0.50000000 0.00000000 0.00000000 p_z
0.50000000 0.00000000 0.00000000 p_x
0.50000000 0.00000000 0.00000000 p_y
0.00000000 0.50000000 0.00000000 p_z
0.00000000 0.50000000 0.00000000 p_x
0.00000000 0.50000000 0.00000000 p_y
0.00000000 0.00000000 0.50000000 p_z
0.00000000 0.00000000 0.50000000 p_x
0.00000000 0.00000000 0.50000000 p_y
```

5 *d* orbitals
9 *p* orbitals

like POSCAR file for VASP
but instead of ionic positions
we include **orbital positions**
*(and names as fourth column)*



DMFTPOSCAR

DMFT.PY

# Part 3: DMFT

## input

```
################   Input parameters for DFT+DMFT calculations   ##################

params = {"Niter":     [12,              "# Number of DMFT iterations"],
          "n_tot":     [25.0,            "# Number of total electrons"],
          "nom":       [6000,            "# Number of Matsubara frequencies"],
          "noms":      [1200,            "# Number of Matsubara frequencies"],
          "nomlog":    [30,              "# Number of Matsubara frequencies"],
          "q":         [[16,16,16],      "# [Nq_x,Nq_y,Nq_z]"],
          "dc_type":   [1,               "# dc type"],
          "U":         [[5.0],           "# Coulomb repulsion (F0)"],
          "J":         [[1.0],           "# Hund's coupling"],
          "Uprime":    [[4.8],           "# Double counting U'"],
          "mu_conv":   [0.0001,          "# The chemical potential convergence condition"],
          "mu_iter":   [50,              "# The chemical potential convergence step"],
          "mu":        [0,               "# The chemical potential (reads from dm.out if 0)"],
          "E_dc":      [0,               "# The double counting energy (reads from dm.out if 0)"],
          "self_dc":   [True,            "# True: Edc=UN(N-1)/2, False: Nd=Nd_f"],
          "Nd_f":      [[8.06],          "# The final Nd"],
          "atomnames": [['Ni','O'],      "# The name of atoms"],
          "cor_at":    [[['Ni1']],       "# Correlated atoms, put degenerate atoms in the same list"],
          "cor_orb":   [[[['d_z2'],['d_x2y2']]], "# DMFT orbitals, other orbitals are treated by HF"],
          "mix_mu":    [0.1,             "# Mixing parameter for mu"],
          "mix_sig":   [0.2,             "# Mixing parameter for Sigma"],
          "mix_dc":    [0.2,             "# Mixing parameter for Edc"],
          "Nd_qmc":    [False,           "# DMFT Nd values are obtained from QMC sampling"],
          "print_at":  [['Ni1','O1'],    "# The local Green functions are printed"],
          "co_at":     [[1.0],           "# The coefficient of Nd for each atom"]
         }
```

dmft_param.py

FLL double counting

to compute a charge ordering measure (don't worry about this)

DMFT.PY

# Part 3: DMFT

## input

```
params_ctqmc = {"exe":        ["~/bin/ctqmc",    "# Path to executable"],
                "Delta":      ["Delta.inp",       "# Input bath function hybridization"],
                "cix":        ["impurity.cix",    "# Input file with atomic state"],
                "mu":         [0,                 "# Chemical potential"],
                "beta":       [100.0,             "# Inverse temperature"],
                "M" :         [20000000,          "# Number of Monte Carlo steps"],
                "nom":        [80,                "# number of Matsubara frequency points to sample"],
                "Nmax":       [1400,              "# maximal number of propagators"],
                "Ntau":       [1000,              "# Ntau"],
                "SampleGtau":  [1000,             "# Sample Gtau"],
                "sderiv":     [0.005,             "# maximal discripancy"],
                "CleanUpdate": [50000,            "# clean update after QMC steps"],
                "aom":        [8,                 "# number of frequency points to determin high frequency tail"],
                "warmup":     [250000,            "# Warmup"],
                "GlobalFlip":  [5000,             "# Global flip"],
                "PChangeOrder":[0.9,              "# Probability to add/remove interval"],
                "TwoKinks":    [0.,               "# Two kinks"],
                "Ncout":      [500000,            "# Ncout"],
                "Naver":      [80000000,          "# Naver"]}
```

dmft_param.py

note: this is per core!

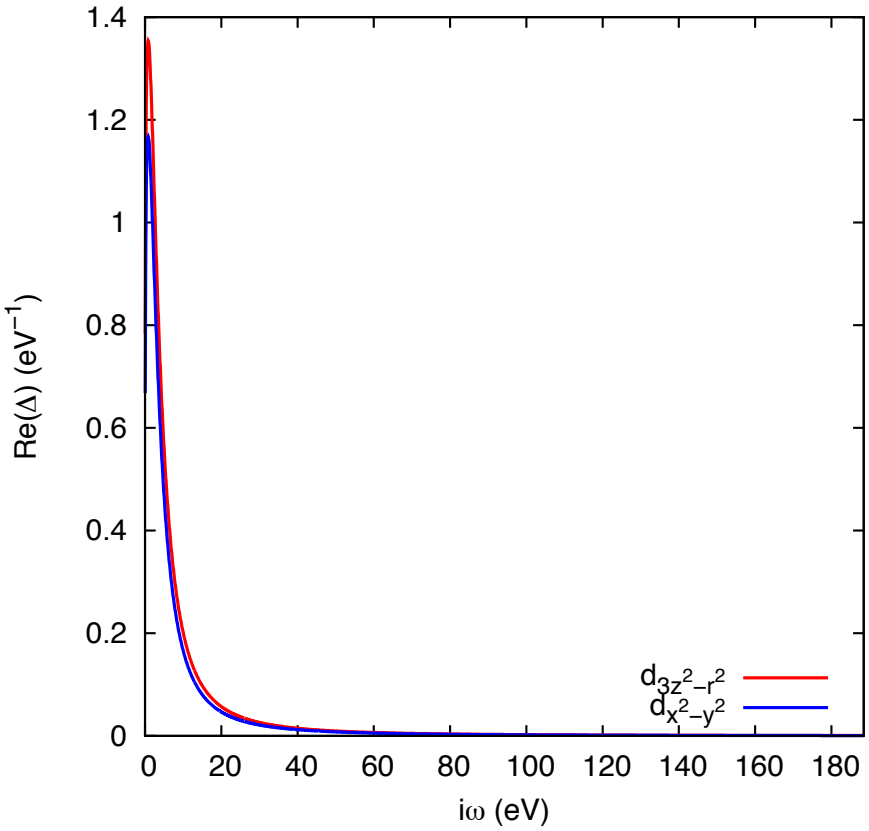
DMFT.PY writes a PARAMS file for CTQMC containing this information

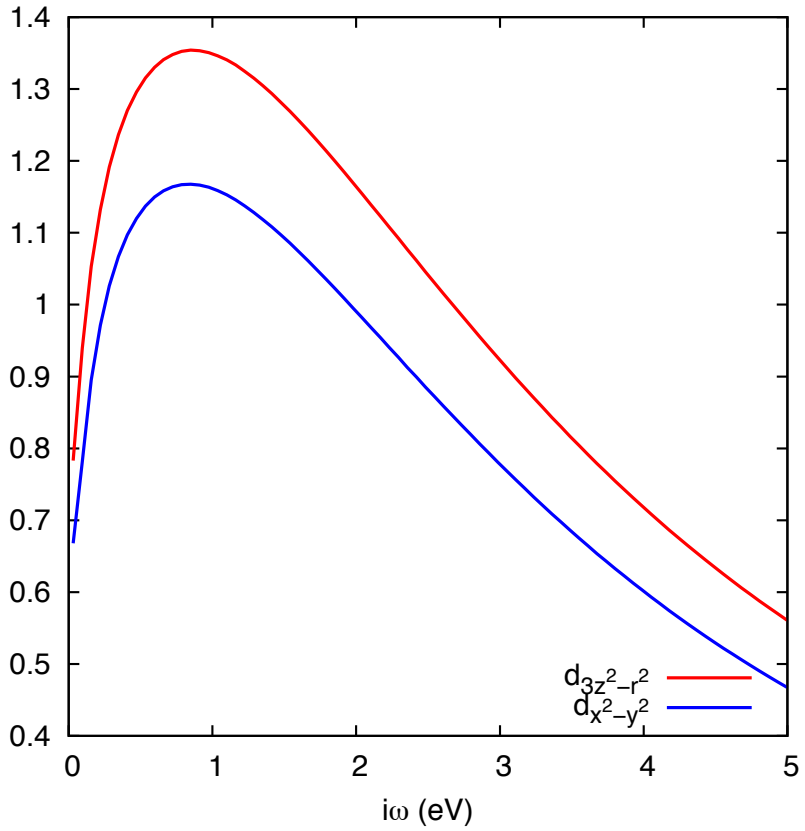
Hyowon has provided reasonable default values, so be careful with changing most things here

DMFT.PY

# Part 3: DMFT

input




Delta.inp

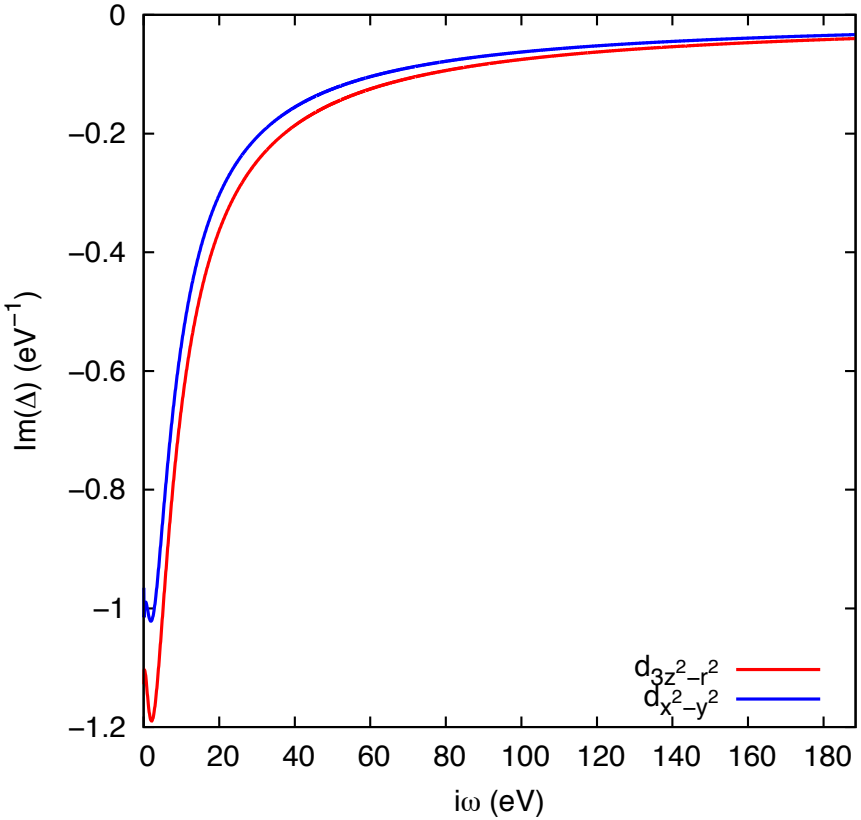
This is generated by DMFT.PY

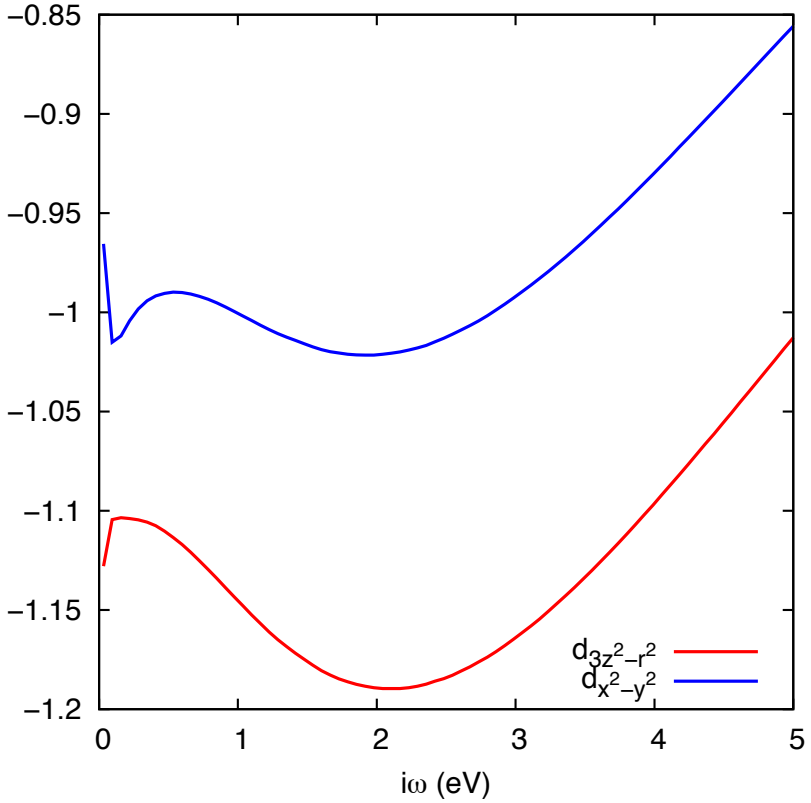
real part

DMFT.PY

# Part 3: DMFT

input




Delta.inp

This is generated by DMFT.PY

imaginary part

DMFT.PY

# Part 3: DMFT

## input

```
# CIX file for ctqmc!
# cluster_size, number of states, number of baths, maximum_matrix_size
1 14 4 2
# baths, dimension, symmetry
0    1 0 0
1    1 0 0
2    1 1 1
3    1 1 1
# cluster energies for non-equivalent baths, eps[k]
0.0 -0.03108101
#   N   K   Sz size
  1   0  0  0.0  1    2   3   5   9    0.0      0.0
  2   1  0  0.5  1    0   4   6   7    0.0      0.0
  3   1  0 -0.5  1    4   0   7  10    0.0      0.0
  4   2  0  0.0  2   12  13   8  11   -1.031563908   0.969401887996      0.0   0.5
  5   1  0  0.5  1    6   7   0   4   -0.03108101      0.0
  6   2  0  1.0  1    0   8   0  12   -3.03108101      0.0
  7   2  0  0.0  2    8  11  12  13   -3.03108101   -1.03108101      0.0   0.5
  8   3  0  0.5  1    0   0   0  14   -5.03108101      0.0
  9   1  0 -0.5  1    7  10   4   0   -0.03108101      0.0
 10   2  0 -1.0  1   11   0  13   0   -3.03108101      0.0
 11   3  0 -0.5  1    0   0  14   0   -5.03108101      0.0
 12   3  0  0.5  1    0  14   0   0   -5.06216202      0.0
 13   3  0 -0.5  1   14   0   0   0   -5.06216202      0.0
 14   4  0  0.0  1    0   0   0   0  -10.06216202      0.0
# matrix elements
  1   2   1  1  1.0
  1   3   1  1  1.0
[…]
```

### impurity.cix

This is generated by DMFT.PY

see [www.physics.rutgers.edu/~haule/681/ctqmc.pdf](www.physics.rutgers.edu/~haule/681/ctqmc.pdf) for documentation

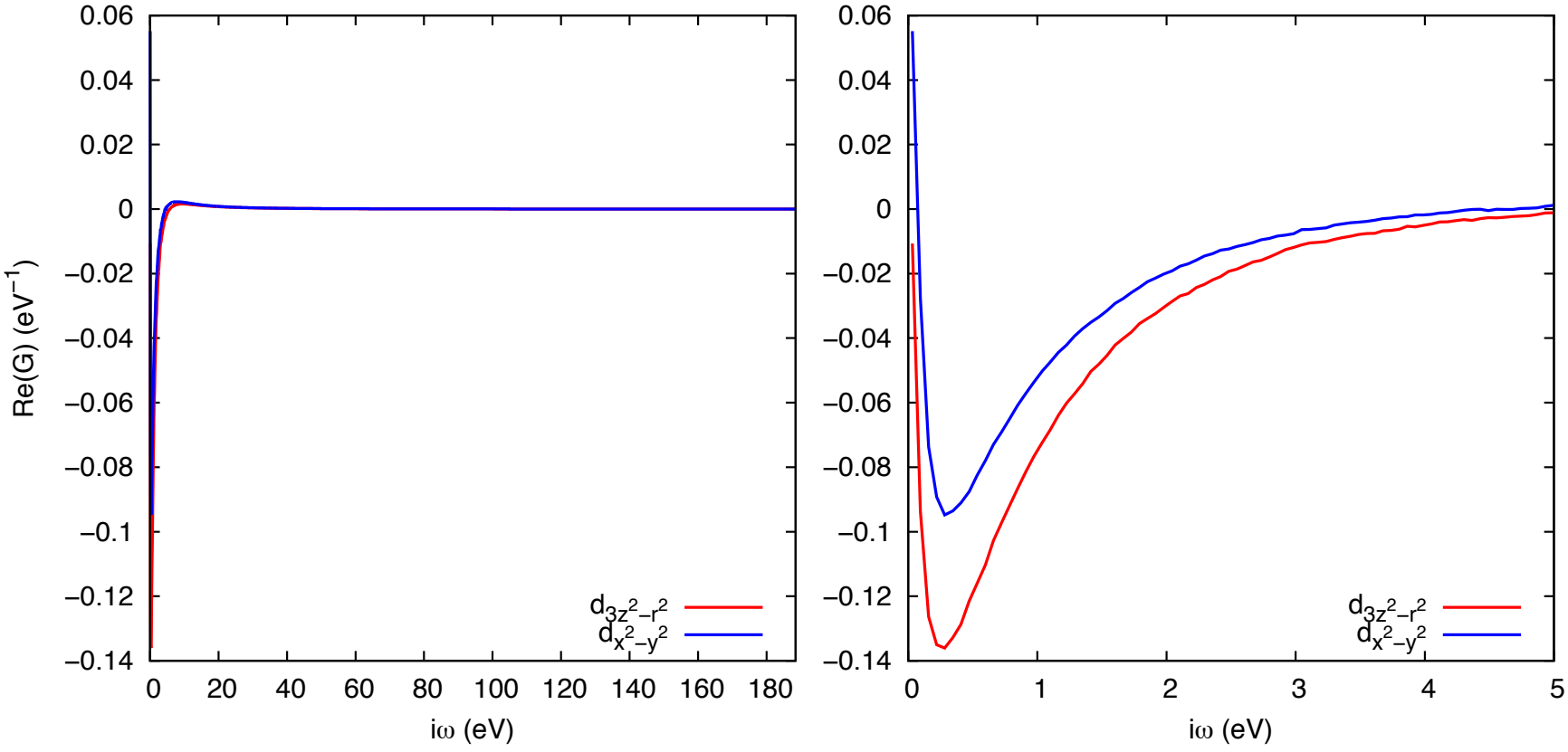
I'm still working to understand this and we will go through it in a later tutorial

DMFT.PY

# Part 3: DMFT

output

ran on 3 16-core Infiniband yeti nodes for around 9 hours, 20 million Monte Carlo steps/core

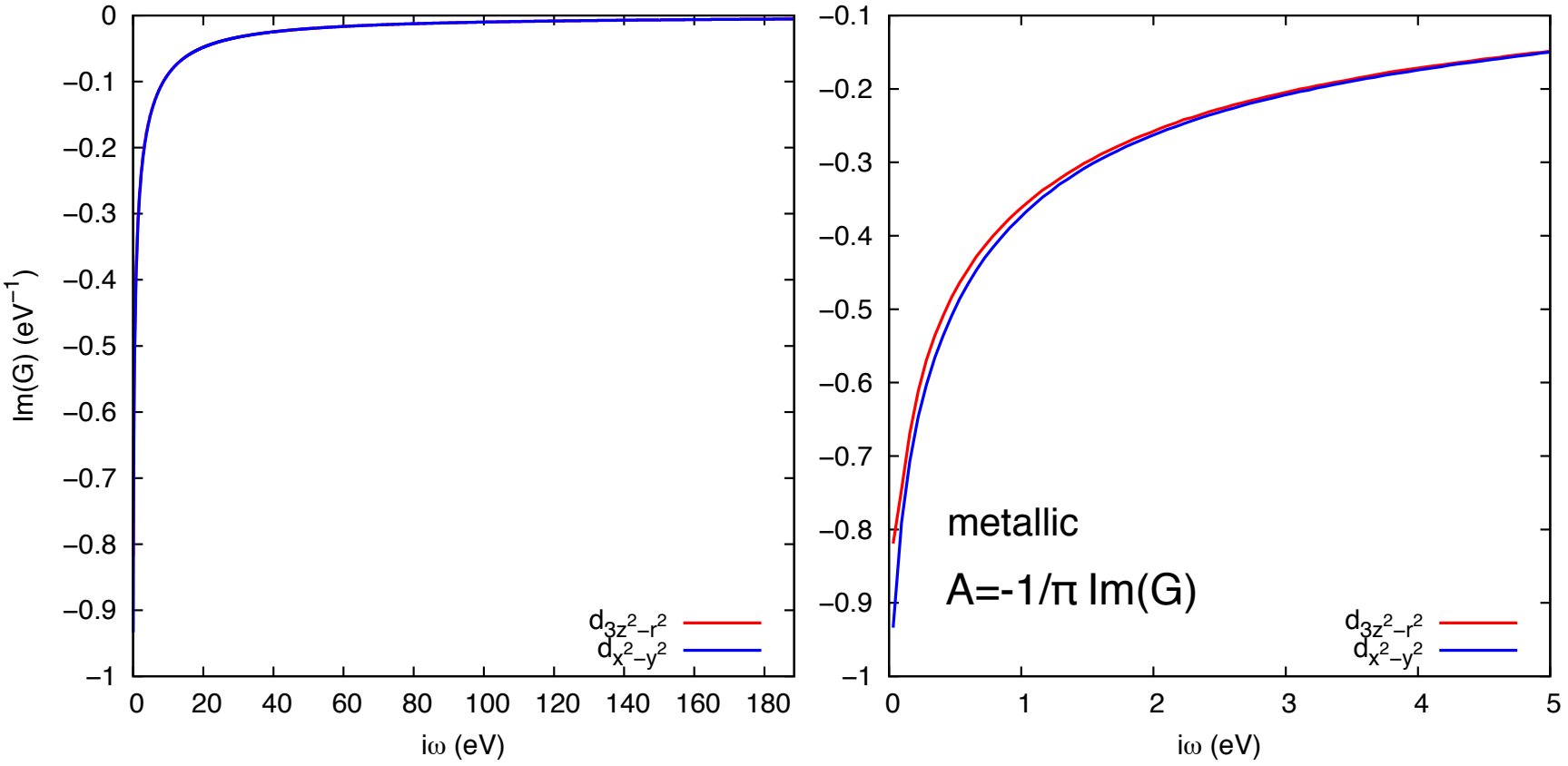



Green function (real part)

DMFT.PY

# Part 3: DMFT

output

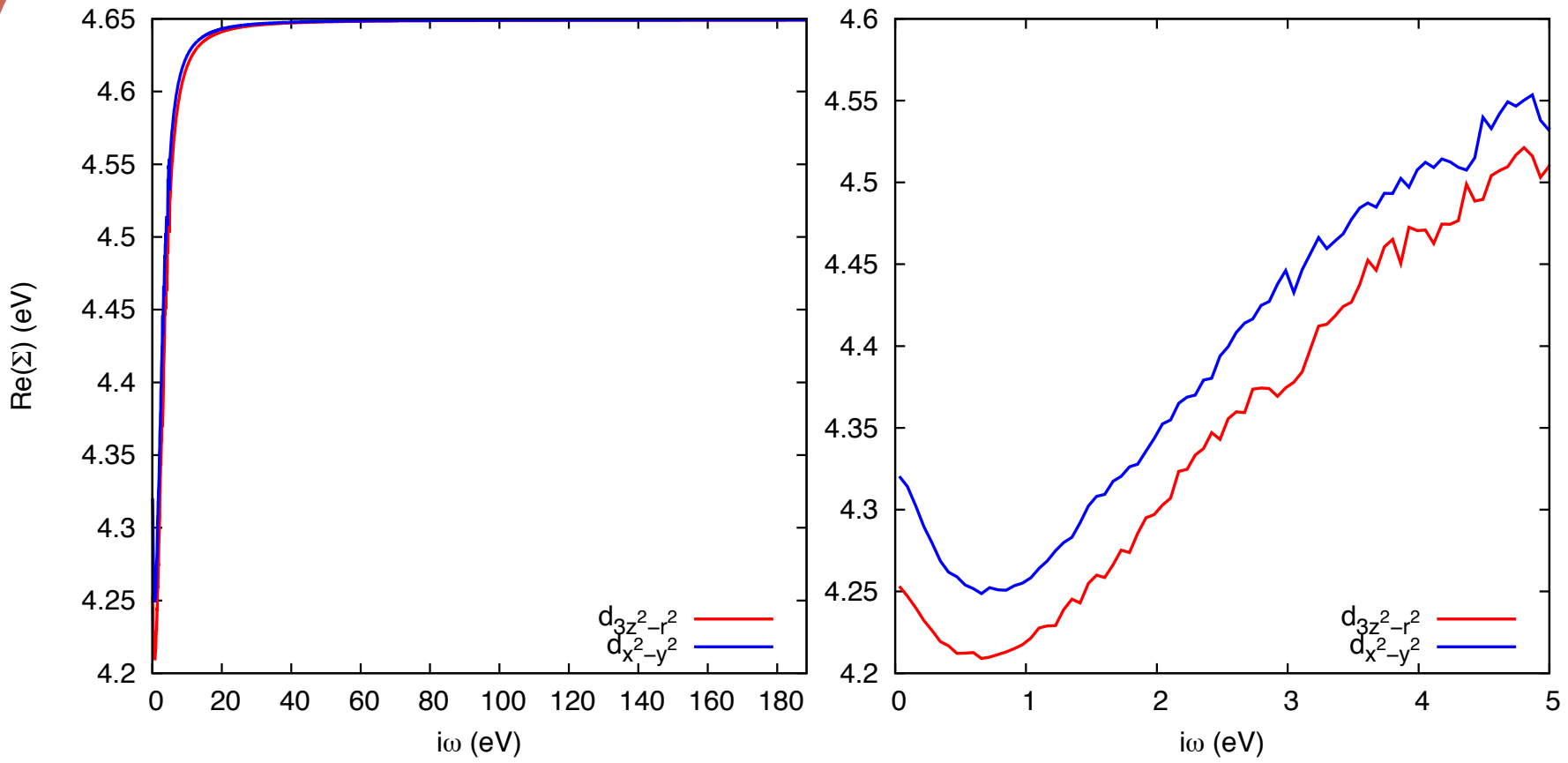



Green function (imaginary part)

DMFT.PY

# Part 3: DMFT

output

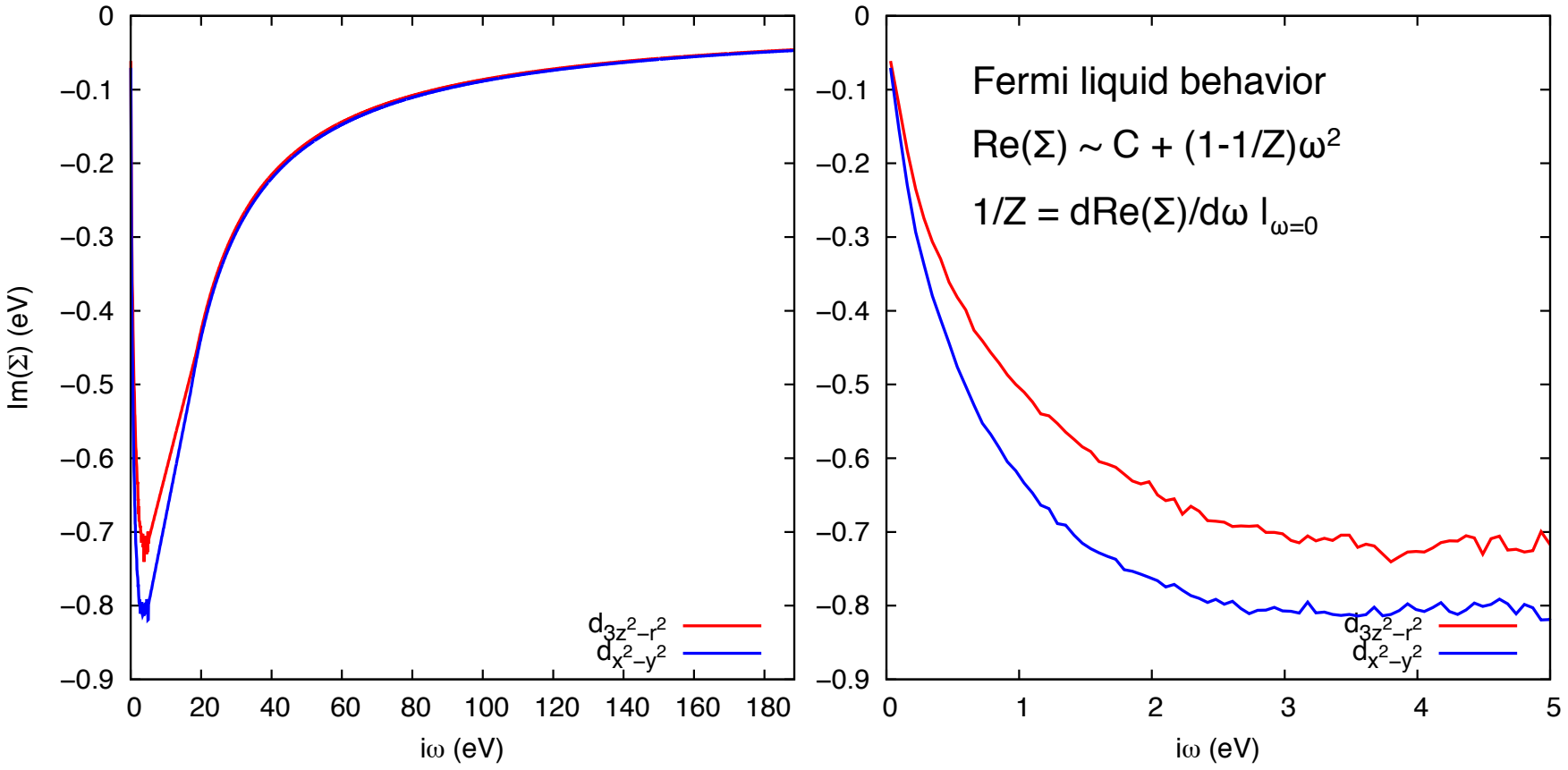



Self-energy (real part)

DMFT.PY

# Part 3: DMFT

output

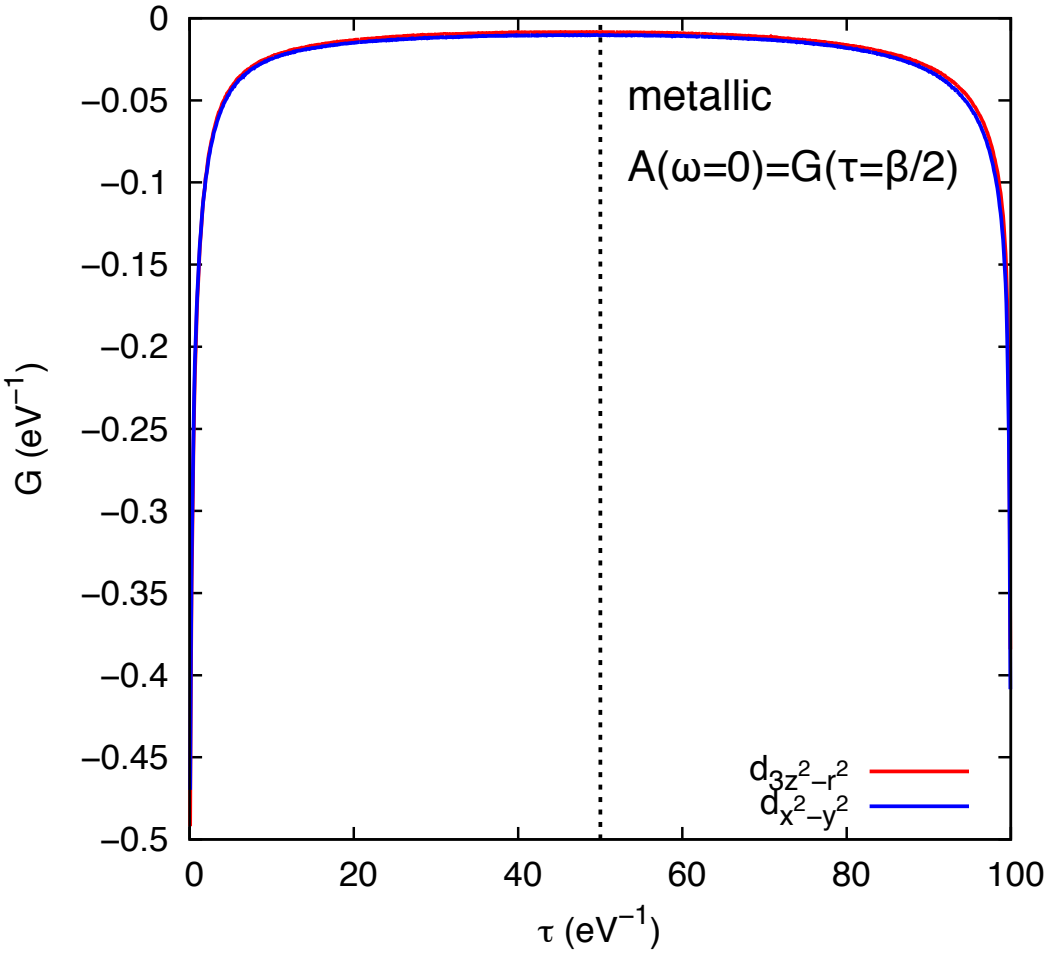



Self-energy (imaginary part)

DMFT.PY

# Part 3: DMFT

output



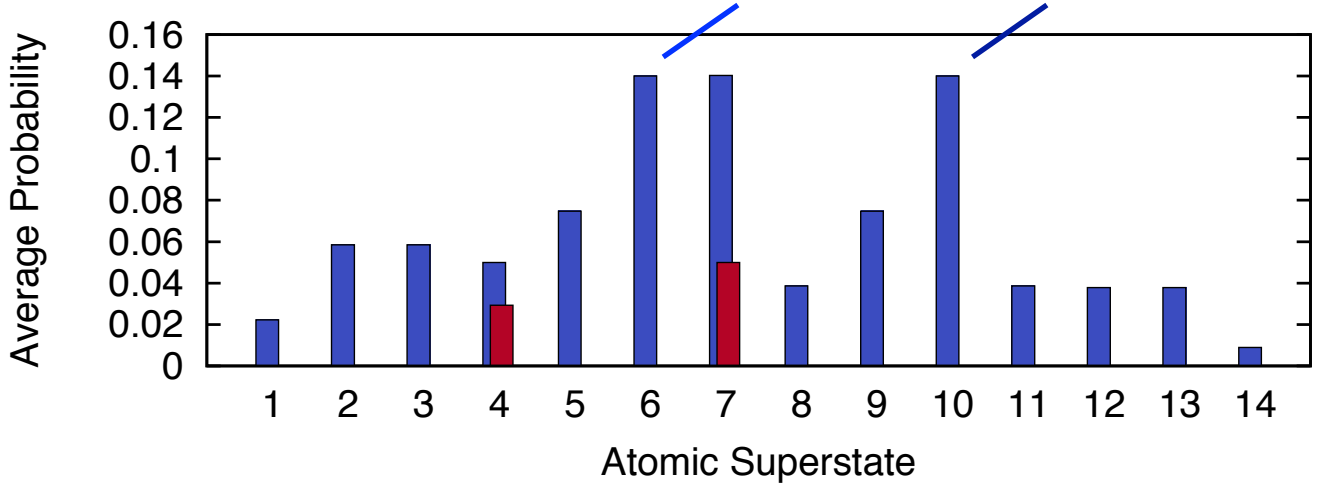

Green function as function of imaginary time

DMFT.PY

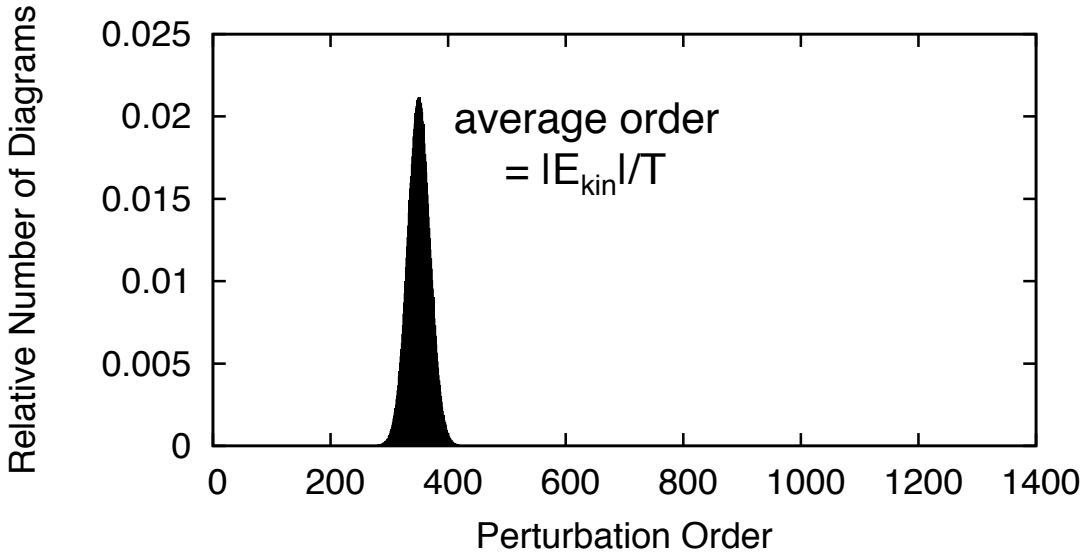
# Part 3: DMFT

output

Configuration Probabilities




1
2




Perturbation Order Histogram

DMFT.PY

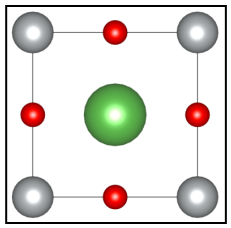
# Part 3: DMFT

output

mu= 7.72924041149 ← chemical potential

E_dc= 18.170835 ← $V_{dc}$

Nd_eg= 1.871483935
Nd_t2g= 5.9501484328 } $N_d$
Nd= 7.8216323678



0.911891 1.976353 1.976353 0.959593 1.997443
1.993684 1.744649 1.995244 1.993685 1.995244
1.744649 1.734936 1.988109 1.988109

} orbital occupancies

dm.out

energetics at each DMFT iteration in Energy_{Kin,Pot,Tot}.dat

local Green functions (including Ni $t_{2g}$ and O *p* states) in G_loc_{Ni1,O1}.out

DMFT.PY

# Further Work

1) Ensure proper convergence of Green function and self-energy

2) Analytical continuation to obtain self-energy on real axis and spectral function (many-body density of states)

3) Study $Li_xCoO_2$ and $Li_xFePO_4$ with this method

# Acknowledgements

Thanks to Hanghui, Jia, Hyowon, and Chris for assistance