Manual for Kristjan Haule's OCA impurity solver

David Jacob
Max-Planck-Institut für Mikrostrukturphysik
Weinberg 2, 06120 Halle, Germany

August 22, 2014

1 NCA and OCA in a nutshell

We consider the general multi-level Anderson impurity model (AIM):

$$\begin{aligned}
\mathcal{H} &= \mathcal{H}_{\rm imp} + \mathcal{H}_{\rm bath} + \mathcal{V}_{\rm hyb} && (1)\\
\mathcal{H}_{\rm imp} &= \sum_{\alpha,\sigma} \varepsilon_\alpha d^\dagger_{\alpha\sigma} d_{\alpha\sigma} + \frac{1}{2} \sum_{\substack{\alpha,\beta,\alpha',\beta'\\ \sigma,\sigma'}} U_{\alpha\alpha'\beta\beta'}\, d^\dagger_{\alpha\sigma} d^\dagger_{\alpha'\sigma'} d_{\beta'\sigma'} d_{\beta\sigma} && (2)\\
\mathcal{H}_{\rm bath} &= \sum_{k,\nu,\sigma} \varepsilon_{k\nu} c^\dagger_{k\nu\sigma} c_{k\nu\sigma} && (3)\\
\mathcal{V}_{\rm hyb} &= \sum_{k,\nu,\sigma} V_{k\nu,\alpha} \left(d^\dagger_{\alpha\sigma} c_{k\nu\sigma} + c^\dagger_{k\nu\sigma} d_{\alpha\sigma} \right) && (4)
\end{aligned}$$

where $\mathcal{H}_{\rm imp}$ is the Hamiltonian of the impurity part of the Anderson model with single-particle energy levels ε_α and the Coulomb interaction given by $U_{\alpha\alpha'\beta\beta'}$. $\mathcal{H}_{\rm bath}$ describes the bath of non-interacting conduction electron bands n with energy dispersion $\varepsilon_{k\nu}$. The hybridization term $\mathcal{V}_{\rm hyb}$ describes the coupling between the impurity and the non-interacting conduction electrons.

The big problem in solving the Anderson model is that the effective Coulomb interaction U between electrons in the impurity is large compared to the hybridzation with the conduction electrons so that a perturbation expansion in terms of the Coulomb interaction is problematic. Instead the basic idea of the Non-Crossing Approximation (NCA) [1] and One-Crossing Approximation (OCA) [3] methods is to treat the hybridization term $\mathcal{V}_{\rm hyb}$ as a perturbation, and develop a digrammatic expansion in terms of $\mathcal{V}_{\rm hyb}$.

We start with an exact diagonalization of the impurity part $\mathcal{H}_{\rm imp}$ of the Anderson Hamiltonian (1). This yields the many-body eigenstates $|m\rangle$ and corresponding eigenenergies E_m of the isolated impurity:

$$\mathcal{H}_{\rm imp} = \sum_m E_m \left|m\right\rangle \left\langle m\right| \tag{5}$$

We now introduce so-called pseudo-particles (PP) representing the many-body eigen states which act on the the PP vacuum (PPV) $|0_{\rm PP}\rangle$:

$$a^\dagger_m \left|0_{\rm PP}\right\rangle = \left|m\right\rangle \text{ and } a_m \left|m\right\rangle = \left|0_{\rm PP}\right\rangle \tag{6}$$

Depending on whether a many-body state $|m\rangle$ represents an even or uneven number of electrons, bosonic or fermionic commutation rules are imposed on the PP operators:

$$\begin{aligned}
[a_m, a^\dagger_m] &= 1 \text{ (Bosons: even number of electrons)} && (7)\\
\{a_m, a^\dagger_m\} &= 1 \text{ (Fermions: uneven number of electrons)} && (8)
\end{aligned}$$

This trick allows us to make use of the machinery of field theory in order to develop a perturbative expansion in terms of the hybridization term below. The physical electron operators $d_{\alpha\sigma}$, $d^{\dagger}_{\alpha\sigma}$ are related to the PP operators by:

$$d_{\alpha\sigma} = \sum_{n,m} (F^{\alpha\sigma})_{nm} a^{\dagger}_{n} a_{m} \tag{9}$$

where $(F^{\alpha\sigma})_{nm} \equiv \langle n| \, d_{\alpha\sigma} \, |m\rangle$ are the matrix elements of the impurity-electron creation operator.

In the PP representation the Anderson model is now given by:

$$\begin{aligned} \mathcal{H} &= \sum_{m} E_m a^{\dagger}_{m} a_m + \sum_{k\nu\sigma} \varepsilon_{k\nu} c^{\dagger}_{k\nu\sigma} c_{k\nu\sigma} \\ &\quad + \sum_{\substack{mn \\ k\nu\alpha\sigma}} \Big(V_{k\nu,\alpha} (F^{\alpha\sigma})_{nm} c^{\dagger}_{k\nu\sigma} a^{\dagger}_{m} a_n + h.c. \Big) \end{aligned} \tag{10}$$

So the impurity part is now diagonal of course. The price we pay is that the hybridization term $\mathcal{V}_{\rm hyb}$ has become more complicated. Moreover, in order to obtain a physical solution additionally the following constraint has to be applied which guarantees the completeness of the impurity many-body Hilbert space:

$$Q \equiv \sum_{m} a^{\dagger}_{m} a_m = 1 \tag{11}$$

The PP charge Q is constraint to one, i.e we can only have one PP at one time. This can be achieved by adding the Lagrangian constraint $\lambda(Q-1)$ to the Hamiltonian where the Lagrange multiplier λ plays the role of a (negative) PP chemical potential. In practice the calculations are done in the grand-canonical PP ensemble. The expectaction value of a physical observable A is calculated in the grand-canonical ensemble and then projected onto the physical subspace with $Q=1$ via the Abrikosov trick:

$$\langle A \rangle = \lim_{\lambda\to\infty} \frac{\langle QA \rangle_G(\lambda_0)}{\langle Q \rangle_G(\lambda_0)} \tag{12}$$

Note that the factor Q in the numerator serves to project out the (unphysical) $Q=0$ subspace. However, for all physical local observables A (acting in the impurity subspace) the expectation value of A in the $Q=0$ subspace is always zero. Hence it suffices to calculate

$$\langle A \rangle = \lim_{\lambda\to\infty} \frac{\langle A \rangle_G(\lambda)}{\langle Q \rangle_G(\lambda)} \tag{13}$$

In the PP picture, the hybridization with the bath electrons given by the last term in eq. (10) becomes the interaction for the PPs. Because of the fermionic and bosonic commutation rules for the PPs, one can now develop a diagrammatic perturbation expansion in the PP interaction. The PP propagators can be written as

$$G_m(\omega) = (\omega - \lambda - E_m - \Sigma_m(\omega))^{-1} \tag{14}$$

where Σ_m is the PP self-energy describing the dynamic interaction with the other PPs induced by the coupling to the conduction electron bath.

The NCA consists in an infinite resummation of self-energy diagrams where conduction electron lines do not cross (hence the name). These are the diagrams shown in the left box in Fig. 1 for a certain PP m. The NCA diagrams describe processes where a single electron (hole) jumps from the bath to the impurity and back thereby temporarily creating a PP with N+1 (N-1) electrons. This corresponds to a self-consistent perturbation expansion to lowest order in the hybridization functions of the impurity levels α

$$\Delta_\alpha(\omega) \equiv \sum_{k,\nu} V^{*}_{k\nu,\alpha} \, g_{k\nu}(\omega) \, V_{k\nu,\alpha} \tag{15}$$

where $g_{k\nu}(\omega) = 1/(\omega^{+} + \mu - \varepsilon_{k\nu})$ is the bare conduction electron propagator. The hybridization function describes the dynamic coupling of an impurity level α to the conduction electrons. Its

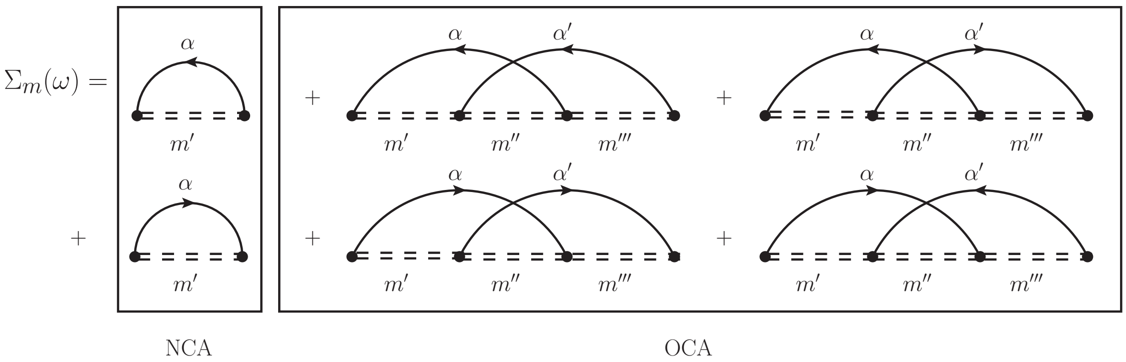

Figure 1: Diagrams for pseudo-particle self-energies in NCA and OCA approximations. full lines correspond to conduction electron propagators coupled to impurity levels α, double dashed lines to full pseudo-particle propagators.

imaginary part $\Delta''_\alpha(\nu)$ gives the single-particle broadening of the impurity level due to coupling to the bath.

Evaluating the NCA diagrams we find the following expression for the NCA self-energy of a PP m:

$$\begin{aligned}\Sigma_m^{\mathrm{NCA}}(\omega) &= -\sum_{m'\alpha\sigma}\Bigg[|F^{\alpha\sigma}_{mm'}|^2\int\frac{d\nu}{\pi}f(\nu)\Delta''_\alpha(\nu)G_{m'}(\omega+\nu)\\ &\qquad + |F^{\alpha\sigma}_{m'm}|^2\int\frac{d\nu}{\pi}f(-\nu)\Delta''_\alpha(\nu)G_{m'}(\omega-\nu)\Bigg] \end{aligned}\tag{16}$$

where $f(\omega)$ is the Fermi function. The first term corresponds to the temporary creation of an electron in the impurity (top NCA diagram in Fig. 1), the second term to the temporary creation of a hole (bottom NCA diagram in Fig. 1). Hence the NCA self-energy is given by a convolution of $\Delta''_\alpha \cdot f$ with the PP propagators $G_{m'}$ of the PPs m' coupled to m via $F^{\alpha\sigma}_{mm'}$. OCA aditionally takes into account diagrams where two bath electron lines cross as shown in the right box of Fig. 1. The algebraic expressions for the OCA self-energies are much more complicated than the NCA ones and involve double convolutions of two hybridization functions (times Fermi function) with three PP propagators. The exact expressions can be found in the literature [3, 5]. Since the self-energy of a PP m depends on the dressed propagators of other PPs m' that interact via $\mathcal{V}_{\mathrm{hyb}}$ with m, the NCA/OCA equations have to be solved self-consistently.

Once the NCA/OCA equations have been solved the real electronic quantities can be calculated from the PP propagators by expanding the real electron operators in terms of PP operators by (9) and using the Abrikosov trick (13). Within NCA, the real electron spectral function ρ_α is obtained from the PP spectral functions as

$$\rho_\alpha(\omega) = \frac{1}{\langle Q\rangle}\sum_{mm'}\int d\varepsilon\, e^{-\beta\varepsilon}[1+e^{-\beta\omega}]|F^{\alpha\sigma}_{mm'}|^2\, A_m(\varepsilon)\, A_{m'}(\omega+\varepsilon)\tag{17}$$

where $A_m(\omega) = -\mathrm{ImG_m}(\omega)/\pi$ is the PP spectral function for PP m. Note that the overall factor $1/\langle Q\rangle$ stems from the Abrikosov trick for calculating physical observables from PP quantities. The PP charge Q can be calculated directly from the PP spectral functions:

$$\langle Q\rangle = \int d\omega\, e^{-\beta\omega}\sum_m A_m(\omega)\tag{18}$$

For a more detailled account of the NCA, OCA and other methods based on a hybridization expansion of atomic states see e.g. Refs. [4, 5].

2 Technical details of the solution of NCA and OCA equations

One problem in evaluating expressions such as (17) is that the PP spectra $A_m(\omega)$ have a threshold behaviour at the lowest PP excitation energy E_0, and diverge in a power-law fashion at the threshold energy E_0 at zero temperature [2]. Moreover, the exponential factors $e^{-\beta\omega}$ diverge strongly for $\omega < 0$ (which is compensated by the threshold behaviour of the PP spectra).

In order to circumvent this problem one introduces new PP spectral quantities

$$\tilde{A}_m = A_m(\omega)/f(-\omega) \text{ and } \tilde{\Sigma}_m(\omega) = \Sigma''_m(\omega)/f(-\omega) \tag{19}$$

The division by the Fermi factors removes the awkward threshold behaviour and also the diverging exponential factors in the equations. Note that $\tilde{\Sigma}$ comprises only the imaginary part of the PP self-energy.

The NCA/OCA equations can now be reformulated in terms of the new tilde quantities:

$$\begin{aligned}
\tilde{A}_m(\omega) &= -\frac{1}{\pi}\frac{\tilde{\Sigma}_m(\omega)}{(\omega+\lambda_0-E_d-\Sigma'_m(\omega))^2+(\Sigma''_m(\omega))^2} \\
\tilde{\Sigma}^{\mathrm{NCA}}_m(\omega) &= \int d\nu \frac{f(\nu)f(-\nu-\omega)}{f(-\omega)} \times \\
&\quad \times \sum_{m'\alpha\sigma} \tilde{A}_{m'}(\omega+\nu)\left(|F^{\alpha\sigma}_{mm'}|^2\Delta''_\alpha(\nu)+|F^{\alpha\sigma}_{m'm}|^2\Delta''_\alpha(-\nu)\right) \qquad (20)\\
\langle Q\rangle &= \int d\omega f(\omega)\sum_m \tilde{A}_m(\omega) \qquad (21)\\
\rho_\alpha(\omega) &= \frac{1}{\langle Q\rangle}\sum_{mm'}\int d\varepsilon\,\frac{f(\nu)f(-\nu-\omega)}{f(-\omega)}|F^{\alpha\sigma}_{mm'}|^2\,\tilde{A}_m(\varepsilon)\,\tilde{A}_{m'}(\omega+\varepsilon) \qquad (22)
\end{aligned}$$

The real part Σ'_m of the PP self-energy Σ_m can be determined from the imaginary part by Kramers-Kronig, thus closing the set of equations. These are the NCA equations which have to be solved self-consistently. The corresponding OCA equations are similar but more complicated as they involve double convolutions of PP spectral functions and hybridization functions. See e.g. Refs. [4, 5] for the exact expressions and more details.

It turns out that the NCA and OCA equations are actually invariant with respect to a total shift $\delta\lambda$ of the frequency argument of *all* PP quantities. In particular, all physical quantities such as the local spectral function ρ_α are not affected by such a shift. Only the PP charge Q changes according to $Q \to Q\exp(-\beta\delta\lambda)$. Therefore λ appearing in the definition of the PP propagator (14) can actually be set to an arbitrary value. Hence we can either fix λ to a certain value and calculate Q, or alternatively, fix Q and calculate λ in each iteration.

Fixing Q in each iteration guarantees that the threshold energy where the PP quantities diverge at zero temeprature is at zero frequency. The resulting λ corresponds to the impurity free energy. For a small number of local states it is best to fix $Q = 1$ in order to achieve convergence. For multi-orbital models with many local states fixing Q to the number of local states is often a good choice.

On the other hand, fixing λ, we can shift zero frequency so that it conincides with the peak position of the main PP spectral function, i.e. the PP with the strongest contribution to the ground state. The main PP often develops an extremely sharp almost delta-like peak (in particular in the Kondo regime) at low temperatures which has to be resolved properly. Therefore an energy mesh with a high resolution is required at the peak position. However, the position of the peak changes in every iteration. Therefore it is convenient to shift the position of the PP peak to zero frequency by a proper choice of λ so that the same mesh can be used in every iteration. Especially in the Kondo regime, this strategy is the only possible for converging the NCA/OCA calculations on a reasonable energy mesh.

3 Input files for the OCA impurity solver

The `oca` program needs four input files: the parameter file (`PARAMS.oca`) containing general calculation parameters, the cix file (default name `cix.dat`) containing information on the pseudo-particles and NCA and OCA diagrams, the pseudo-particle self-energy input file (default name `Sigma.000`) containing the initial guess for the pseudo-particle self-energy, and the hybridization function/bath spectral function input file which defines the hybridization function for each impurity level.

Additionally, a few more input files are generally needed in order to generate the above input files. A file specifying the impurity Hamitlonian is read by the `gencix.py` script in order to generate the cix file. Frequency meshes have to be generated for resolving pseudo-particle functions and real electron functions.

3.1 Information about the impurity and pseudo-particles: the CIX file

The so-called CIX file (Correlated IndeX) contains the information about the pseudo-particles and the NCA and OCA diagrams taken into account in the calculation. It can be generated by running the `gencix.py` script. The gencix.py script requires information on the impurity Hamiltonian $\mathcal{H}_{\mathrm{imp}}$ (number of impurity levels, energy levels ε_α, Coulomb repulsion U), and the valencies of the impurity to be considered in the OCA calculation. This information can be passed either directly as command line arguments to the `gencix.py` script in the form of Python definitions, or by writing an input file containing these variable definitions and passing the file name as a command line argument.

Variable	type (default)	description
`nlevels`	`integer (1)`	number of impurity levels
`Eimp`	`real list ([-1.])`	single-particle energies ϵ_α for each impurity level
`U0`	`real (2.)`	intra-level Coulomb repulsion U
`U1`	`real (1.)`	inter-level Coulomb repulsion U'
`JH`	`real (0.5)`	Hund's rule coupling J
`Ucfile`	`string ('none')`	File name for generic Coulomb interaction U_{ijkl} to be read from file
`n`	`integer list ([0,1,2])`	valencies of impurity to be considered for NCA/OCA diagrams
`Ncentral`	`integer list ([1])`	central valency for OCA diagrams
`qOCA`	`integer (1)`	whether to calculate OCA diagrams
`Eoca`	`real (10.)`	energy cutoff (relative to lowest pseudo-particle) for taking into account OCA diagrams
`mOCA`	`real (0.001)`	OCA diagram weight above which OCA diagrams are taken into account

The Coulomb interaction $\mathcal{V}_c$ of the impurity shell can be defined in two ways: either a model interaction is chosen (default), comprising direct repulsion $U \equiv U_{\alpha\alpha\alpha\alpha}$ between electrons in the same impurity level, direct repulsion $U' \equiv U_{\alpha\beta\alpha\beta}$ between electrons in different impurity levels $\alpha \neq \beta$, and Hund's rule coupling $J_H \equiv U_{\alpha\beta\beta\alpha}$:

$$\mathcal{V}_c = U \sum_\alpha \hat{n}_{\alpha\uparrow}\hat{n}_{\alpha\downarrow} + U' \sum_{\alpha<\beta} \hat{n}_\alpha \hat{n}_\beta - J_H \hat{S}_d^2 + J_H(N_d - D - N_d^2/4) \tag{23}$$

where $\hat{n}_{\alpha\sigma} = d^\dagger_{\alpha\sigma} d_{\alpha\sigma}$, $n_\alpha = \hat{n}_{\alpha\uparrow} + \hat{n}_{\alpha\downarrow}$, $N_d = \sum_\alpha n_\alpha$, D is the number of doubly occupied impurity levels and S_d is the total impurity spin.

Alternatively, one can read in a completely general Coulomb matrix U_{ijkl} by specifying the filename of corresponding file with the variable `Ucfile`. The file format for the Coulomb interaction file is the following: each line specifies one matrix element U_{ijkl} in the form $i\,j\,k\,l\,U_{ijkl}$. The indexes run from 1 to `nlevels`.

Let's consider a simple example: the one-level Anderson impurity model (1AIM). The impurity Hamiltonian is given by

$$\mathcal{H}_{\rm imp} = \epsilon_d \hat{n}_d + U \hat{n}_{d\uparrow}\hat{n}_{d\downarrow} \tag{24}$$

We want to consider the Kondo regime where $n_d \approx 1$. Thus we consider valencies $n_d = 0, 1, 2$ and the central valency is $n_d = 1$. We choose $U = 4$ and $\epsilon_d = -2$. The corresponding input parameters for the gencix.py script are (can be found in file `imp.dat` in `examples/1AIM/input` subdirectory):

```
nlevels=1        # Number of impurity levels
Eimp= [-2.0]     # single-particle energy for each impurity level...
U0=4.0           # intra-orbital Coulomb repulsion U
n=[0,1,2]        # Valencies of impurity to be considered..
Ncentral=[1]     # Central valency for OCA diagrams
qOCA = 1         # Whether to calculate OCA diagrams
Eoca = 10.0      # Energy cutoff for taking into account OCA diagrams
mOCA = 0.001     # OCA diagram weight cutoff
```

We can now generate the corresponding cix-file by typing `gencix.py imp.dat` at the command line. The gencix.py script performs an exact diagonalization of the impurity Hamiltonian $\mathcal{H}_{\rm imp}$ specified by the variables `Eimp` and `U0`, and for the valencies specified by the variable `n`. In the next step the matrix elements $F^{\alpha}_{mn} = \langle m|\, d_\alpha \,|n\rangle$ for the resulting many-body eigenstates $|n\rangle$ are calculated.

From these the matrix elements of NCA and OCA self-energy diagrams for the pseudo-particles are precomputed whereby we sum over the degeneracies of the PPs and impurity levels. A matrix element for an NCA diagram is thus given by:

$$C^{\bar{\alpha}}_{\bar{m}\bar{n}} = \sum_{\alpha\in\deg(\bar{\alpha}),m\in\deg\bar{m},n\in\deg\bar{n},\sigma} |F^{\alpha\sigma}_{mn}|^2 \tag{25}$$

With this precomputed weight, a self-energy diagram for a PP m due to coupling to PPs $\bar{n}$ contributing to the total NCA self-energy for that PP can be written as:

$$\tilde{\Sigma}^{(\bar{n})}_m(\omega) = \frac{C^{\bar{\alpha}}_{\bar{m}\bar{n}}}{N_{\rm deg}(m)} \times \int d\nu \frac{f(\nu)f(-\nu-\omega)}{f(-\omega)} \tilde{A}_n(\omega+\nu)\Delta''_\alpha(\nu) \tag{26}$$

Hence total self-energy for PP m is given by $\tilde{\Sigma}_m = \sum_{\bar{n}} \Sigma^{(\bar{n})}_m$.

And similarly for the OCA diagrams we have the sum over degeneracies of PPs and impurity levels for a product of four F matrices:

$$D^{\bar{\alpha}\bar{\beta}}_{\bar{m}\bar{n}\bar{o}\bar{p}} = \sum_{\alpha,\beta,m,n,o,p\in\deg,\sigma,\sigma'} F^{\beta\sigma'}_{mp} F^{\alpha\sigma}_{po} F^{\beta\sigma'\dagger}_{on} F^{\alpha\sigma\dagger}_{nm} \tag{27}$$

The resulting cix-file `cix.dat` looks like this:

```
# Input file for OCA impurity solver. nlevels= 1 Eimp= [-2.0] mOCA= 0.001...
1 2  3 0 # Number of impurity levels, their degeneracy and number of local...
  #     N0 Mtot  deg        Eatom  #b  #f
  0   0.00    0    1     0.000000   1   0      2.000000 x 1
  1   1.00    1    2     0.000000   1   1      2.000000 x 2      2.000000 x 0
  2   2.00    2    1     4.000000   0   1      2.000000 x 1
# OCA diagrams, information is (pp0,pp1,pp2,pp3) (b1,b2) fact , where pp...
  0   1   2   1     0  0 2.0    # [0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
```

The first line is a pure comment line (ignored by `oca` program) containing some of the input parameters specifying the impurity Hamiltonian. In the second line the number of non-degenerate impurity levels (1), the degeneracy of each of those (2 because of spin), the number of non-degenerate local valence states (3) which correspond to the pseudo-particles, and number of core states (0) is specified. Third line is again a pure comment line (ignored by `oca`) with a descriptor for each of the columns in the following lines.

In the next 3 lines the properties of each pseudo-particle (=local valence states) are given together with the NCA diagram weights. The first column is the index of the PP (starting with 0), the next column(s) give the occupation(s) for each non-degenerate impurity level (`N0`=impurity level 0, `N1`=impurity level 1 etc.), The following column (`Mtot`) gives the total number of electrons (i.e. the valency) of this local PP. The column entitled `deg` gives the degeneracy of the PP, and the next column entitled `Eatom` gives the atomic energy E_m of the PP m. Note that only the Coulombic part of the atomic energy E_m is given. The single particle part given by the energy levels ε_α has been substracted as it is added later within the `oca` program (see below).

We see that we have three PPs of which PP $m = 1$ is doubly degenerate This corresponds to the four atomic states $|0\rangle$, $|\uparrow\rangle$, $|\downarrow\rangle$ and $|\uparrow\downarrow\rangle$ of the 1AIM. The singly occupied states $|\uparrow\rangle$ and $|\downarrow\rangle$ are degenerate (in the absence of a magnetic field or spin-polarization of the bath).

In the rest of the line all NCA self-energy diagrams Σ_m for the PP m are specified in the following way. The column '`#b`' gives the number of 'backward' diagrams, i.e. the number of diagrams where a conduction electron moves backward in time, temporarily creating an additional electron in the impurity. Correspondingly, the column '`#f`' gives the number of 'forward' diagrams where an electron is temporarily destroyed in the impurity. Then for each 'backward' diagram the pre-computed weight C^α_{mn} (where α denotes the impurity level) 'times' the PP n which is temporarily created is given. Then the same for each 'forward' diagram. Note if there are more than one non-degenrate impurity levels, then for each level backward and forward diagrams are specified corresponding to creation or annihilation of electrons in that level. Here is a PP description line in the cix-file for the non-degenerate 2-level Anderson model:

```
#  N0 N1 Mtot deg    Eatom #b #b  #f #f
0  1. 0.    1   2 0.000000  1  2   0  0  2.0 x 3  3.0 x 2  1.0 x 4
```

After the lines specifying the PPs and NCA diagrams, the OCA diagrams are given in a different format. First, a comment line explaining the format of the OCA diagram descriptions is given. Then in each line an OCA matrix element is specified in the format:
$m\,n\,o\,p\,D^{\alpha\beta}_{mnop}$
Note that a cyclic permutation of PP indices m, n, o, p and impurity levels α, β yields the same matrix elements. These matrix elements are equivalent and are only stored once in the cix file.

3.2 Parameter file: General calculation parameters

While the PP properties and the NCA and OCA diagrams are completely determined by the impurity model and thus do not change during the calculation of a certain model, other calculation parameters such as the temperature or the mixing parameter for the self-consistent solution of the NCA/OCA equation might change. These type of parameters is specified in the parameter file `PARAMS.oca` file. Also the impurity level energies ϵ_α are specified here, as one might want to explore effect of shifts of the impurity levels (induced by a gate for example). Shifting of the impurity levels only changes the atomic energies E_m of the PPs but does not affect the other properties such as the matrix elements $F^{\alpha\sigma}_{mn}$, and hence neither the precomputed NCA and OCA matrix elements.

The following table gives an overview of all calculation parameters that can be specified in the `PARAMS.oca` file:

1. Input and output file name specifications

Variable	default val.	description
`Sig`	`'Sigma.000'`	file name of the input Self-energies
`Ac`	`'None'`	file name of the input bath function if only the spectral function is given
`Delta`	`'None'`	file name of the input bath function if a retarded Delta is given
`out`	`'.'`	file name of the output directory
`gloc`	`'gloc.out'`	file name of the output Green's function
`sig`	`'sig.out'`	file name of the output Self-energy
`SigOut`	`'Sigma.000'`	file name of the output pseudo-particle Self-energy
`AlocOut`	`'Aloc.imp'`	file name of the output Physical Spectral function
`cix`	`'cix.dat'`	file name of the input file containing information about bands and their degeneracy

1. Calculation parameters

Variable	default val.	description
`Ed`	`[-2.]`	impurity energy levels ϵ_d
`U`	`0.`	direct Coulomb repulsion U (same for all orbitals) can also be added here in addition to what is already given in cix-file
`T`	`0.2`	Temperature
`Q`	`1`	Default average Q in grand-canonical ensamble
`prt`	`1`	Wether to print intermediate results
`prtSF`	`1`	Wether to print out spectral functions of pseudo-particles
`alpha`	`0.5,`	Fraction of new PP self-energy to be used in the next iteration
`max_diff`	`1e-6`	Convergence criterion: difference in PP functions between consecutive steps
`max_steps`	`1`	Maximum number of steps in NCA/OCA self-consistency
`followPeak`	`-3`	Mode to determine λ and Q: -3: fix Q to number of local states, calculate λ -2: fix λ to value of `StartLambda`, calculate Q -1: fix Q to value of `Q`, calculate λ $m \geq 0$: fix λ such that PP m has diverging peak at zero frequency, calculate Q
`StartLambda`	`-1`	where to start searching for the lambda0
`EndLambda`	`1`	where to stop searching for the lambda0
`dLambda`	`0.1`	step in searching for the lambda
`MissDopSt`	`-100`	missing doping due to projection and finite mesh starting at MissDopSt
`N_ac`	`50`	number of points to add for calculating Delta to be used in core calculation
`dom_ac`	`0.5`	distance between points that are added to the mesh for core calculation
`acore`	`0`	wether to add core contribution to auxiliary self-energies
`pcore`	`1`	wether to add core contribution to physical spectral function
`renorm_core`	`0`	renormalization of core
`renorm`	`1`	renormalization of spectral function
`OnlyCore`	`0`	only core is added to physical spectral function. It must already exist
`ExcludePP`	`-1`	list of pseudo particles to exclude from calculation of physical spectral function

Note that the `Ed` variable refers to non-degenerate impurity levels only. If we have for example two impurity levels with the same energy, then `Eimp=[-2,-2]` (gencix.py input) but `Ed=[-2]` in PARAMS.oca!

Here is the `PARAMS.oca` file for the single-level Anderson model calculation found in the `examples/1AIM/input` subdirectory:

```
Ac=Ac.inp         # Name of bath spectral function input file
gloc=gloc.out     # Name of local Green's function output file
Sig=Sigmapp.inp   # Name of PP Self-enery initial guess input file
sig=sig.out       # Name of electron slef-energy output file
#
Ed=[-2.0]         # energies of non-degenerate impurity level(s)
followPeak=1      # mode to determin lambda
```

```
max_diff=0.0001   # maximum difference between steps
max_steps=20      # maximum impurity steps
alpha=0.5         # Fraction of new self-energy to be mixed with old one
T=0.1             # Temperature
```

Here we have set `followPeak=1` which means that λ will be fixed in each iteration such that zero frequency is set to the position of the diverging PP peak of PP $m = 1$.

3.3 Pseudo-particle self-energy input file

Finally, OCA needs an initial guess for the PP self-energy. This initial guess is defined in the input file specified by the `Sig` parameter of the `PARAMS.oca` file. Each line contains a frequency, and the real and imaginary parts of the PP self-energies at that frequency for all PPs specified in the out.cix file, i.e. line i in the file is given by:

$$\omega_i \qquad \mathrm{Re}\Sigma_0(\omega_\mathrm{i}) \qquad \mathrm{Im}\Sigma_0(\omega_\mathrm{i}) \qquad \mathrm{Re}\Sigma_1(\omega_\mathrm{i}) \qquad \mathrm{Im}\Sigma_1(\omega_\mathrm{i}) \qquad \ldots$$

The initial guess can actually be empty. In that case the OCA program generates its own initial guess. However, the self-energy input file *cannot* be empty. It must at least contain the frequency mesh for the PP functions! Resolving the PP spectra properly is one of the key issues when solving the NCA/OCA equations. A proper frequency mesh needs a high resolution at zero frequency and sometimes also at some positive finite frequency in order to proper resolve sharp peaks of certain PPs. Such a mesh can be generated by using the `gaumesh.py` Python script described in the next section.

3.4 Resolving pseudo-particle spectra on an energy mesh: the gaumesh.py script

In order to properly resolve PP peaks in the PP spectra one has to generate a mesh with a non-homogeneous frequency spacing with a high resolution around zero frequency and sometimes also for some positive finite frequencies, if other PPs at higher energies also yield a sharp peak. The `gaumesh.py` Python script generates a mesh from a superposition of Gaussian functions centered at different frequencies and with different widths for the mesh density. Integration of the such generated Gaussian mesh density thus yields the frequency mesh with high resolution near the center of the Gaussian functions. Hence the frequency mesh is actually build up by a superposition of error functions centered at different frequencies and with different widths.

Similar to the `gencix.py script`, the `gaumesh.py` script takes Python variables as an input for specifying the widths and the positons of the Gaussians directly from the command line, or alternatively from a file whose name is passed to the `gencix.py` script as a command line parameter.

Here is an example for a mesh input file (`ppmesh.inp`) which can be found in the `examples/mesh` subdirectory:

```
dx0 =  [2e-7, 2e-5, 2e-4, 2e-2, 0.2,  ]
fwhm = [2e-5, 2e-3, 2e-2, 2.0,  20.0, ]
x0 =   [0.0,  0.0,  0.0,  0.0,  0.0,  ]
xmin = -20.0
xmax =  20.0
```

The variable `x0` specifies a list of frequencies where the Gaussian functions will be centered. In this case all Gaussians are centered at zero frequency but have different widths and resolutions near the center. The variable `fwhm` is a list of real numbers that determine the width (full width at half maximum) of each Gaussian. The list of real numbers specified by `dx0` determines the maximal resolution of each Gaussian mesh near its center. Hence we have a superposition of five

Gaussian meshes all centered at zero with growing widths and decreasing resolutions. As a rule of thumb the number of points in a single Gaussian mesh is approximately given by `2 * fwhm / dx0`. Hence each Gaussians in `ppmesh.inp` contributes roughly 200 points to the mesh adding up to a total of roughly 1000 points (the exact number 1060 points, see `meshpp.dat`). Finally, xmin and xmax specify the range of mesh points, i.e. here we generate a mesh from -20 to 20.

If we want to resolve a further PP at some frequency, say $\omega = 0.1$ we can simply add one or more Gaussians at this point:

```
dx0 =  [2e-7, 2e-5, 2e-4, 2e-2, 0.2,  0.1  ]
fwhm = [2e-5, 2e-3, 2e-2, 2.0,  20.0, 2e-3 ]
x0 =   [0.0,  0.0,  0.0,  0.0,  0.0,  2e-5 ]
xmin = -20.0
xmax =  20.0
```

Note that also the real electron functions (local Green's function $G_d(\omega)$, spectral function $\rho_d(\omega)$, self-energy $\Sigma_d(\omega)$) have to be resolved on a mesh. The real electron mesh is defined by the hybridization function (or equivalently by the bath spectral function) input file discussed in the next section.

3.5 Hybridization function input file and resolution of real electron functions

The hybridization function $\Delta_\alpha(\omega)$ describes the coupling of the impurity to the conduction electron bath, and is treated as a perturbation to the impurity Hamiltonian $\mathcal{H}_{\text{imp}}$. Also the mesh on which the hybridization function is resolved defines the mesh for *all* real electron functions such as the local Green's function $G_d(\omega)$, spectral function $\rho_d(\omega)$, self-energy $\Sigma_d(\omega)$.

There are two ways of reading in the hybridization function. If the variable `Delta` in the `PARAMS.oca` script is defined, the full hybridization function is read from the file with file name specfied by the `Delta` variable. Each line contains the frequency, and real and imaginary parts for each non-degenerate impurity level α at that frequency. Alternatively, the so-called bath spectral function can be read instead. The bath spectral function is given by the imaginary part of the hybridization function as $A_c(\omega) = -\Delta''(\omega)/\pi$. The real part of the hybridization function is then obtained by Kramers-Kronig. The bath spectral function is read if the variable `Ac` is defined in the `PARAMS.oca` file. The format is: first column are frequencies, in the following columns the bath spectral function for each non-degenerate impurity level.

In order to resolve the real electron functions properly also the mesh for the hybridization functions/bath spectral functions, we also need an imhomogenous mesh with a high resolution around zero frequency in order to resolve the Kondo peak. Therefore the mesh defined by the hybridzation function/bath spectral function input file is pretty much similar to the mesh for resolving the PP functions. An example is given in the `mesh` subdirectory: `acmesh.inp` and `meshac.dat`. Note that in order to capture the tails of the PPs properly, the PP mesh usually needs to extend farther than the real electron mesh.

4 Self-consistent OCA calculation and output files

In each iteration of the NCA/OCA self-consistent calculation output is written to log file(s) (`oca_log.XXX`, one file for each processor in a parallel calculation, where XXX denotes the process ID in the MPI parallel environment). Typical output during one step of the self-consistency for the single-impurity Anderson model defined above:

```
  2.) KramarsKronig
  2.) DeterminSpectralFunctions ; lambda = -2.092206683951923
       Q = 2.001835728656129
       n0=0.009360830670693009
```

```
     n1=0.9812783386585935
     n2=0.009360830670713539
      Q is here equal to 2.001835728656129
2.) nd:  1.000000000000021
  Norm of Spectral functions:
   0:  1.0005   1:  1.0007   2:  1.0005
0:  1.0000
2.) SelfConsistentStep

2.) Difference between steps:  0.006614
```

The '`2.)`' denotes the 2nd iteration in this case. First, the real part of the PP self-energies is generated from the imaginary part $\tilde{\Sigma}_m$ by Kramers-Kronig (output in 1st line). Then the PP spectral functions $\tilde{A}_m$ are determined via eq. (20). Then Q and λ are determined depending on the mode specified by `followPeak` in PARAMS.oca. Here `followPeak=1` so that λ is fixed such that PP $m = 1$ has the diverging peak at zero frequency, and Q is calculated. The resulting λ and Q are given in 2nd and 3rd line, respectively. Below Q the contribution of each PP to the total ground state of the system is printed:

```
n0=0.009360830670693009
n1=0.9812783386585935
n2=0.009360830670713539
```

So the strongest contribution comes from the PP $m = 1$ which in this case are the doubly degenerate states $|\uparrow\rangle$ and $|\downarrow\rangle$. This is the main PP which develops a sharp peak at zero at low temperature. These contributions should sum up to 1.

Next, the real electron charge N_d is determined and printed in the 8th line:
`2.)  nd:  1.000000000013894`.
After that the norm of the PP spectra is calculated to ensure that they are properly resolved. A color code is used additionally to highlight which PPs are very well resolved (green), moderatley well resolved (yellow), or very badly resolved (purple). The PPs with the strongest contribution should always be 'green' in order to converge the NCA/OCA calculations. For all other PPs 'yellow' is usually enough. 'Purple' PPs are almost always problematic.

After the norm of the PPs, the occupations of different impurity levels is printed (in blue). Here we have only one (doubly degenerate) impurity level $\alpha = 0$ which has occupation of exactly 1. Finally, the new PP self-energy is calculated according to eq. 20, and mixed with the old one according to parameter `alpha` defined in `PARAMS.oca`. The difference between the new PP self-energy and old one is calculated to determine whether the self-consistent calculation is converged. This difference is printed in the last line.

In each iteration the following files are also generated: The PP self-energy (file name `Sigma.XYZ` where XYZ is the iteration number), the PP spectral function (file name `Sigma.XYZ~`), and the current real electron spectral density (`A00.current`).

Once the program terminates, either by convergence of the NCA/OCA self-consistency, or by exceeding of `max_steps`, the following files are generated: the final electron spectral function (default name `Aloc.imp`), the electronic local Green's function (default name `gloc.out`), and the electronic self-energy (default name `sig.out`). Also the final PP self-energies and spectra are written to files (default names: `Sigma.000` and `Sigma.000~`).

In subdirectory `examples/1AIM/output` output files for calculations at different temperatures can be found.

References

[1] P. Coleman, Phys. Rev. B **29**, 29 (1984)

[2] J. Kroha and P. Wölfle, Acta Phys. Pol. B **29**, 3781 (1998)

[3] K. Haule *et al.*, Phys. Rev. B **64**, 155111 (2001)

[4] G. Kotliar *et al.*, Rev. Mod. Phys. **78**, 865 (2006)

[5] K. Haule *et al.*, Phys. Rev. B **81**, 195107 (2010)