Solve finite-bandwidth Eliashberg equations

Eliashberg theory

The linearized local Eliashberg equations for the density of states $N(\varepsilon) = N(0)\,\Theta(E - |\varepsilon|)$ are

$$\tilde{\omega}_n = \omega_n + 2T \sum_m^{|\omega_m|<\omega_N} \arctan\left[\frac{E}{\tilde{\omega}_m}\right] \lambda_{n-m},$$
$$\phi_n = 2T \sum_m^{|\omega_m|<\omega_N} \arctan\left[\frac{E}{\tilde{\omega}_m}\right] \frac{\phi_m}{\tilde{\omega}_m} (\lambda_{n-m} - \mu^*_N),$$

where

$$\frac{\lambda}{\lambda_n} = 1 + \left[\frac{\nu_n}{\omega_\mathrm{E}}\right]^2,$$
$$\frac{1}{\mu^*_N} = \frac{1}{\mu} + \frac{2}{\pi} \int_0^{\frac{E}{\omega_N}} \frac{\mathrm{d}x}{x} \arctan(x).$$

$\omega_n = (2n+1)\pi T$ $[\tilde{\omega}_n]$ and $\nu_n = 2n\pi T$ are [renormalized] fermionic and bosonic Matsubara frequencies and ϕ_n is proportional to the order parameter at the critical point $(T, \lambda, \mu, \omega_\mathrm{E}, E)$.

Parameters

`T`	T	K	temperature
`l`	λ	1	electron-phonon coupling
`u`	μ	1	Coulomb repulsion
`w`	ω_E	eV	Einstein phonon frequency
`E`	E	eV	half the electronic bandwidth
`W`	ω_N	ω_E	cutoff frequency

Functions

`def critical(variable='T', epsilon=1e-3, **parameters)`

solves the Eliashberg equations varying the parameter indicated by `variable`, which may be T, λ, μ, ω_E or E, leaving the others fixed. Its given value is used as initial guess, which must not vanish, its critical value is returned. `epsilon` is the self-consistency threshold.

`def Tc(l, u, w, E, A=1.20, B=1.04, C=0.62, **ignore)`

returns the critical temperature according to McMillan's formula,

$$T_\mathrm{c} = \frac{\omega_\mathrm{E}}{A} \exp\left[\frac{-B(1+\lambda)}{\lambda - C\lambda\mu^* - \mu^*}\right] \quad \text{with} \quad \frac{1}{\mu^*} = \frac{1}{\mu} + \log\left[\frac{E}{\omega_\mathrm{E}}\right].$$

`def Eliashberg(T, l, u, w, E, W, rescale=True, **ignore)`

returns the maximum eigenvalue of the kernel of the equation for ϕ_n, which is less than, equal to or greater than unity in the normal, critical or superconducting state, respectively. If `rescale` is `False`, μ is used in place of μ^*_N.

`def residue(x)`

returns $1/\mu^*_N - 1/\mu$ as a function of E/ω_N.

`def eigenvalue(matrix)`

returns the maximum eigenvalue of the given matrix.