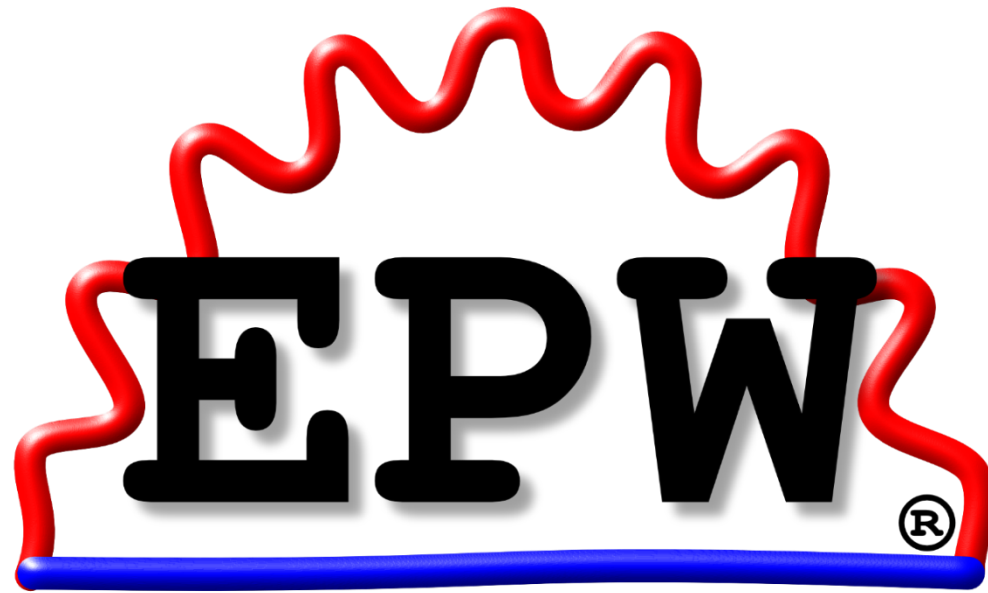
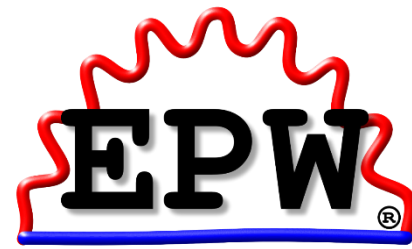




Samuel Poncé, University of Oxford
Feliciano Giustino, University of Oxford

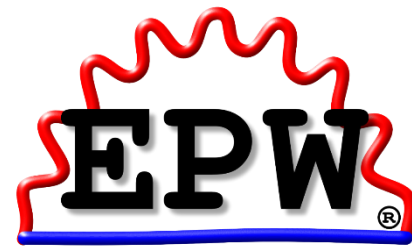
Overview of the code



- Electron-Phonon using Wannier interpolation (EPW)
- Rely on el-ph matrix elements produced by QE
- Interpolate them on a dense k/q grid

- The code can be used to compute
 - ▣ Electron and phonon linewidths & lifetimes
 - ▣ ZPR and temperature dependence
 - ▣ Eliashberg $\alpha^2 F$, $\alpha^2 F_T$ and electronic $A_{nk}(\omega, T)$ spectral function $\alpha^2 F$
 - ▣ Anisotropic superconducting gap Δ_{nk}
 - ▣ Transport properties

Overview of the code

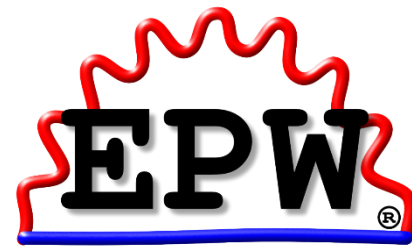


- EPW rely on some QE routines but need not to modify them

Espresso/EPW/bin
 /examples
 /src
 /tests
/Modules
/Phonon
/PW
/test-suite
/wannier90-2.0.1

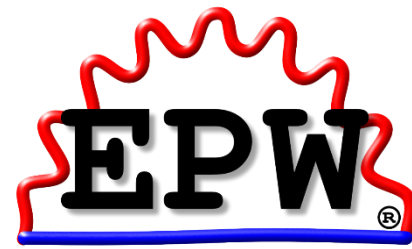


Recent developments



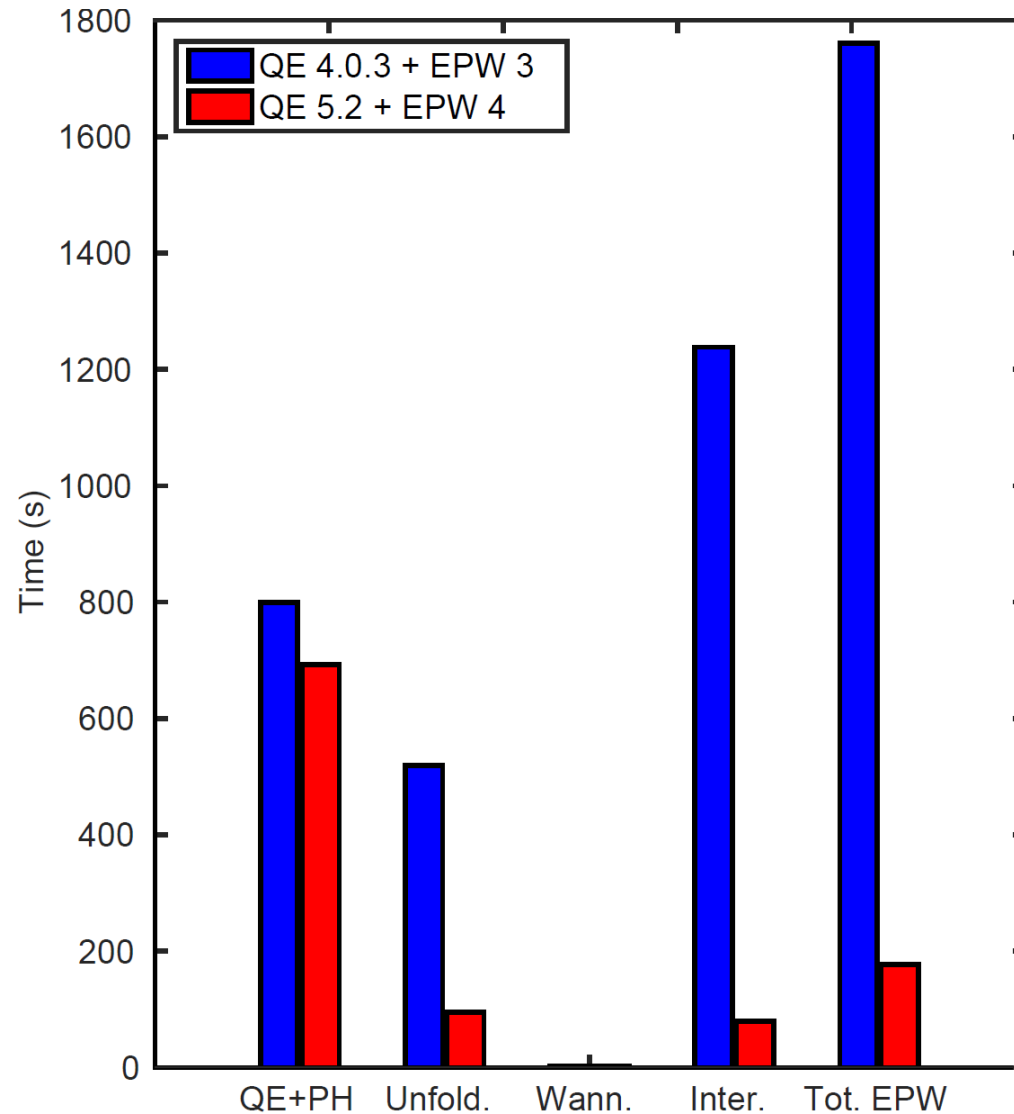
- EPW is now fully compatible with the latest QE trunk and the new Wannier90 2.0.1 lib
- Optimization & cleaning of the code
- Time-reversal symmetry is included
- Polar divergence correctly interpolated
- Spin-orbit coupling included (non-magnetic)
- We plan on releasing EPW in QE in the next 6 weeks

Limitation of the code

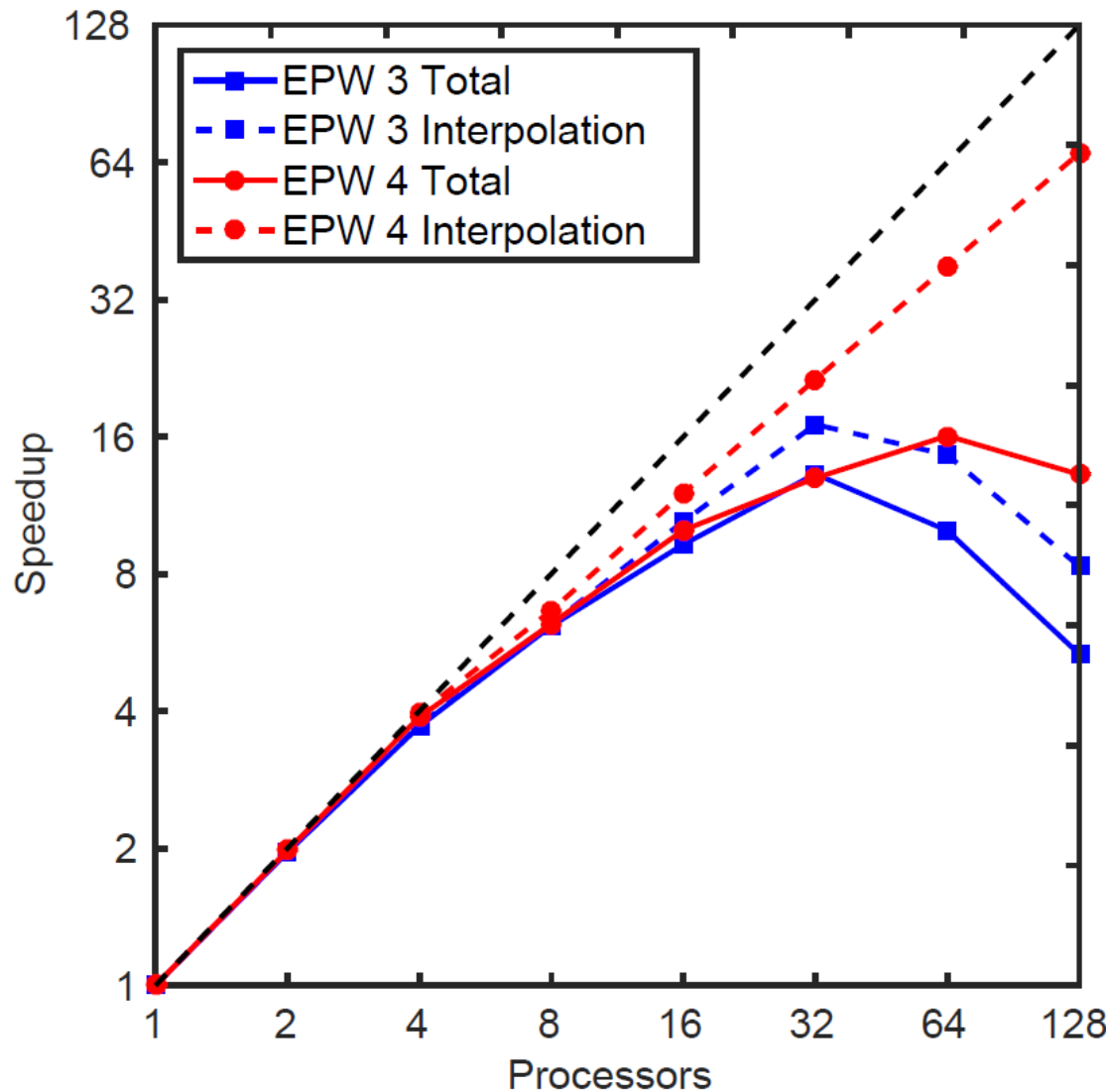


- NC psp only
- No magnetization
- No G -vector parallelization
- Nothing beyond LDA/GGA [not in ph.x]
- No +U [not in ph.x]

Parallelization & Speedup (SiC)



Parallelization & Speedup



Buildbot test farm

- EPW is tested for continuous integration using nightly Buildbot test suites.
- We have 4 hardware slaves: 2 Intel Xeon X5345@2.33Ghz, Opteron P265@1.8Ghz and Intel Core 2 Duo@2.8Ghz
- We plan on adding a Mac

[Home](#) - [Waterfall](#) | [Grid](#) | [T-Grid](#) | [Console](#) | [Builders](#) | [Recent Builds](#) | [Buildslaves](#) | [Changesources](#) - [JSON API](#) - [About](#)

Waterfall

[waterfall help](#)

		waterfall help														
last build		EPW-farm2_qcc520 build successful	EPW-farm2_intel13 build successful	EPW-farm2_intel15 build successful	EPW-farm2_nag6 build successful	EPW-farm2_pqi15 build successful	EPW-farmer3_qcc493 build successful	EPW-farmer3_intel15 build successful	EPW-farmer_qcc485 failed checkout_qe shell_1	EPW-farmer_qcc485_serial build successful	EPW-farmer_intel12 build successful	EPW-farmer_intel13 failed make_pw shell_1	EPW-farmer_intel15 build successful	SGW-farm2_qcc520 build successful	SGW-farm2_intel13 build successful	SGW-farm2_intel15 build successful
current activity		idle	idle	idle	idle	idle	idle	idle	idle	idle	idle	idle	idle	idle	idle	idle
GMT	changes	EPW-farm2_qcc520	EPW-farm2_intel13	EPW-farm2_intel15	EPW-farm2_nag6	EPW-farm2_pqi15	EPW-farmer3_qcc493	EPW-farmer3_intel15	EPW-farmer_qcc485	EPW-farmer_qcc485_serial	EPW-farmer_intel12	EPW-farmer_intel13	EPW-farmer_intel15	SGW-farm2_qcc520	SGW-farm2_intel13	SGW-farm2_intel15
									'make clean' failed stdio							
									update failed stdio							
									'make veryclean' stdio							
									Build 18							
14:33:35																
14:12:42																
14:11:58																
14:11:05																
Thu 07 Jan 2016 10:19:34	sponsor															
Sun 20 Dec	sponsor															

Buildbot test farm

Slaves	OS	Compiler	Flags	Parallel	Library
farmer1	Ubuntu 14.04.3	gcc 4.8.5	-O3 -g -x f95-cpp-input	-	internal
farmer1	Ubuntu 14.04.3	gcc 4.8.5	-O3 -g -x f95-cpp-input	openmpi 1.8.8	internal
farmer1	Ubuntu 14.04.3	intel 12.1.2	-O2 -assume byterecl -g -fpp	openmpi 1.8.8	MKL
farmer1	Ubuntu 14.04.3	intel 13.1.3	-O3 -xHost -assume byterecl -assume buffered_io -fpp	openmpi 1.8.8	MKL + fftw 3.3.4
farmer1	Ubuntu 14.04.3	intel 15.0.3	-O2 -assume byterecl -g -fpp	openmpi 1.8.8	MKL + fftw 3.3.4
farmer2	CentOS 7.1.1503	gcc 5.2.0	-O3 -g -x f95-cpp-input	openmpi 1.8.8	internal
farmer2	CentOS 7.1.1503	pgf90 15.7	-fast	mpich 3.1.3	fftw 3.3.4
farmer2	CentOS 7.1.1503	intel 13.0.1	-O3 -xHost -assume byterecl -assume buffered_io -fpp	mpich 3.1.4	MKL + fftw 3.3.4
farmer2	CentOS 7.1.1503	intel 15.0.3	-O3 -g -x f95-cpp-input	Intel mpi 5.1	MKL + fftw 3.3.4
farmer2	CentOS 7.1.1503	Nag 6.0	-O0 -kind=byte -dcfuns -mismatch	-	internal
farmer3	openSUSE 13.2	gcc 4.9.3	-O3 -g -x f95-cpp-input	mpich 3.1.3	fftw 3.3.4
farmer3	openSUSE 13.2	intel 15.0.3	-O3 -xHost -assume byterecl -assume buffered_io -fpp	mvapich2 2.2a	MKL + FFT MKL

User friendly: tests & tutorials



- EPW has a fully dedicated website: <http://epw.org.uk>
- Tutorials, tests, documentations and theory are available
- Video tutorials are planned (screen recording + sound)

User friendly: tests & tutorials

EPW Documentation/Documentation - Mozilla Firefox

Inbox (3,8... x UCL-IMC... x QEForge > Pr... x Behind Th... x BuildBot: ... x Advanced Qu... x http://...W.html x ARCHER »... x RGB Color ... x EPW Docume... x QEForge > Pr... x Scientific Linux x

epw.org.uk/Documentation/Documentation

SLinux

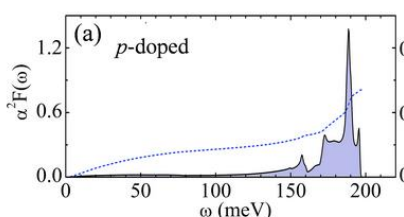
Most Visited ▾ Solid_State ▾ Lab ▾ Social ▾ Bank ▾ Buy info ▾ Music ▾ location-history-visua... ▾ Setting Up Docker an... ▾ Mind Reference: Usin... ▾ Printing | IT Services ... ▾ ScholarOne Manuscri...



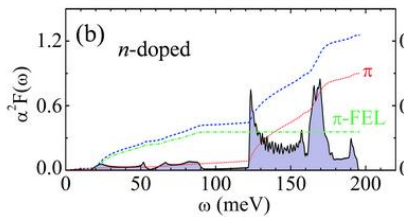
DOCUMENTATION

Here you can find the documentation on the Inputs variables of EPW, the theory behind it and some tutorials (navigate using the right-hand side menu).

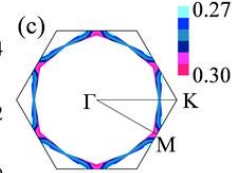
(a) *p*-doped



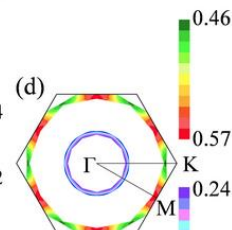
(b) *n*-doped



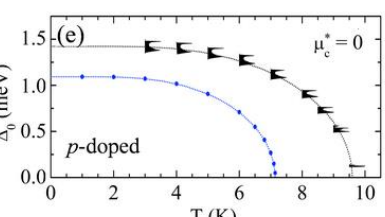
(c)



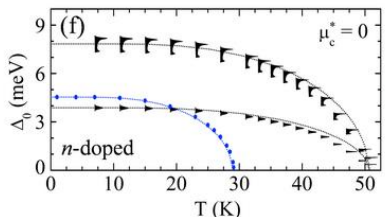
(d)



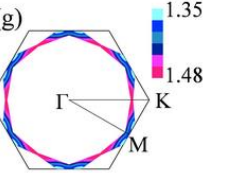
(e)



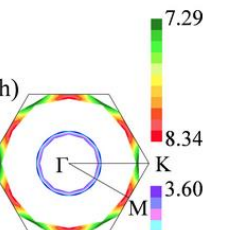
(f)



(g)



(h)



PROJECT

About

Documentation

Inputs

Theory

Tutorial

Acknowledge

FAQ

Papers using EPW

Contact

SOURCE CODE

Download and Install

Subversion

Test Farm

Troubleshoot

Edit

EPW dependences on QE

- To not re-invent the wheel, we import modules from QE:
- Modules/
 - ▣ mp*, **wavefunctions**, ions_base, cell_base, noncol, io_files, io_global, fft_base, recvec, control_flags
- S3DE/iotk/src/iotk_module
- PW/src/
 - ▣ pwcom, start_k, **symm_base**
- PHonon/PH/
 - ▣ **phcom**, **smallgq**, **set_small_group_of_q**
- PP/src/sym_band, cryst_to_car
- Wannier90-2.0.1/src
 - ▣ parameters, wannier_libs, wannierise

Test case: B-doped diamond

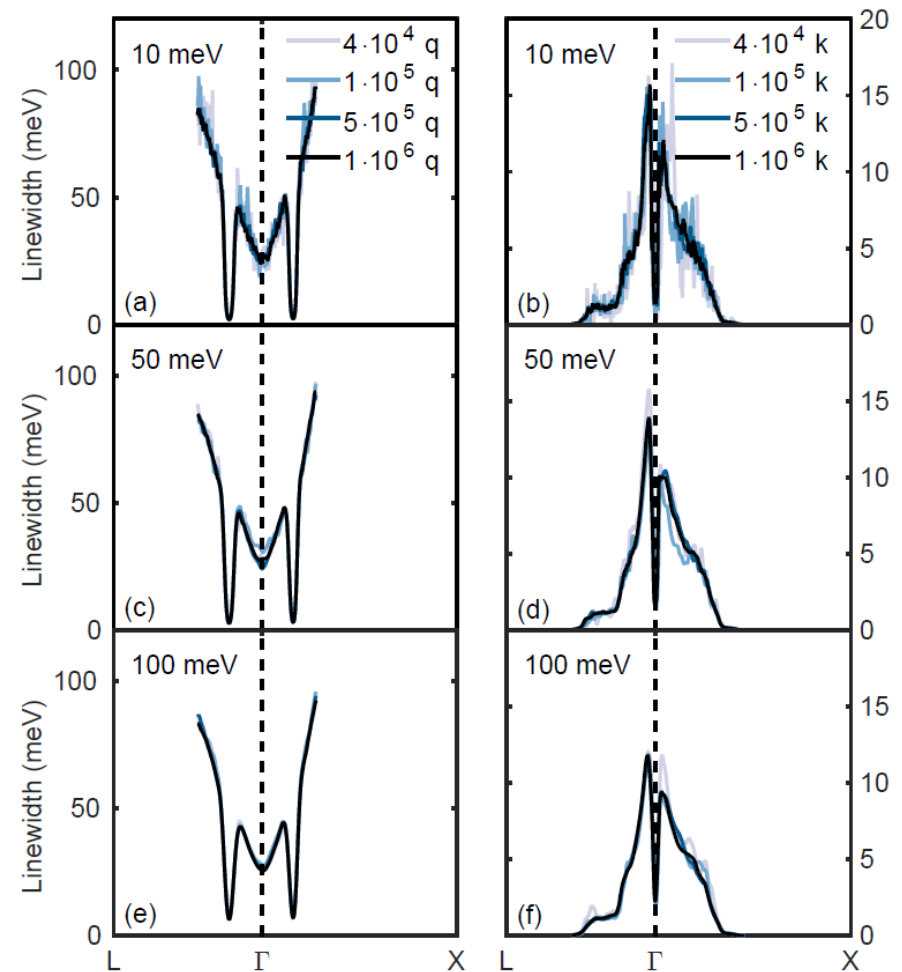
□ Electron linewidth = $\Im \Sigma_{nk}$

$$\Sigma_{nk}(\omega, T) = \sum_{mv} \int_{BZ} \frac{d\mathbf{q}}{\Omega_{BZ}} |g_{mn,v}(\mathbf{k}, \mathbf{q})|^2$$

$$\left[\frac{n_{\mathbf{q}v}(T) + f_{m\mathbf{k}+\mathbf{q}}(T)}{\omega - \varepsilon_{m\mathbf{k}+\mathbf{q}} + \omega_{\mathbf{q}v} + i\delta} + \frac{n_{\mathbf{q}v}(T) + 1 - f_{m\mathbf{k}+\mathbf{q}}(T)}{\omega - \varepsilon_{m\mathbf{k}+\mathbf{q}} - \omega_{\mathbf{q}v} + i\delta} \right]$$

□ Phonon linewidth = $\Im \Pi_{nk}$

$$\Pi_{\mathbf{q}v}(\omega, T) = \sum_{mn} \int_{BZ} \frac{d\mathbf{k}}{\Omega_{BZ}} |g_{mn,v}(\mathbf{k}, \mathbf{q})|^2 \frac{f_{n\mathbf{k}}(T) - f_{m\mathbf{k}+\mathbf{q}}(T)}{\omega - \varepsilon_{m\mathbf{k}+\mathbf{q}} - \omega_{\mathbf{q}v} + i\delta}$$

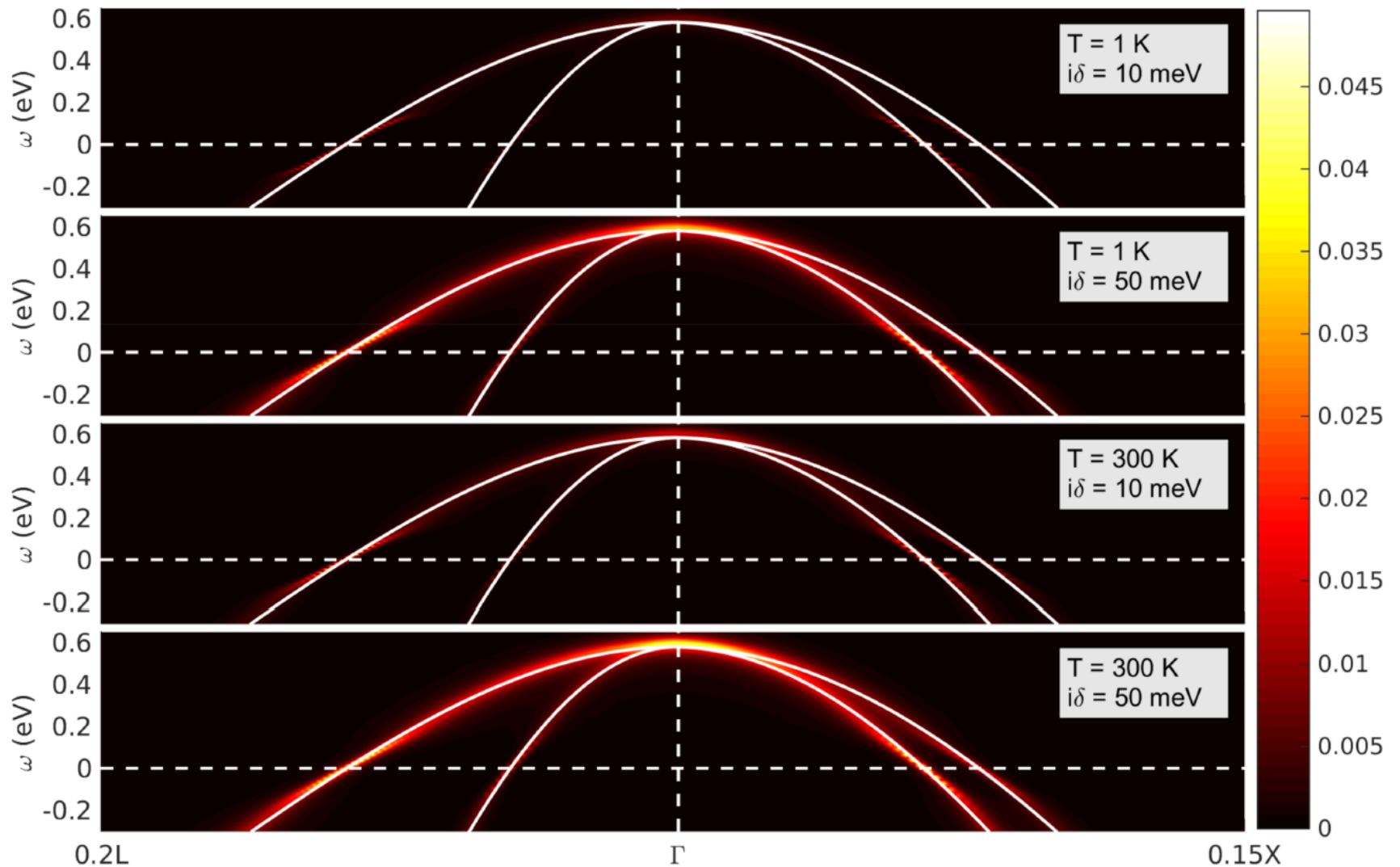


Test case: B-doped diamond

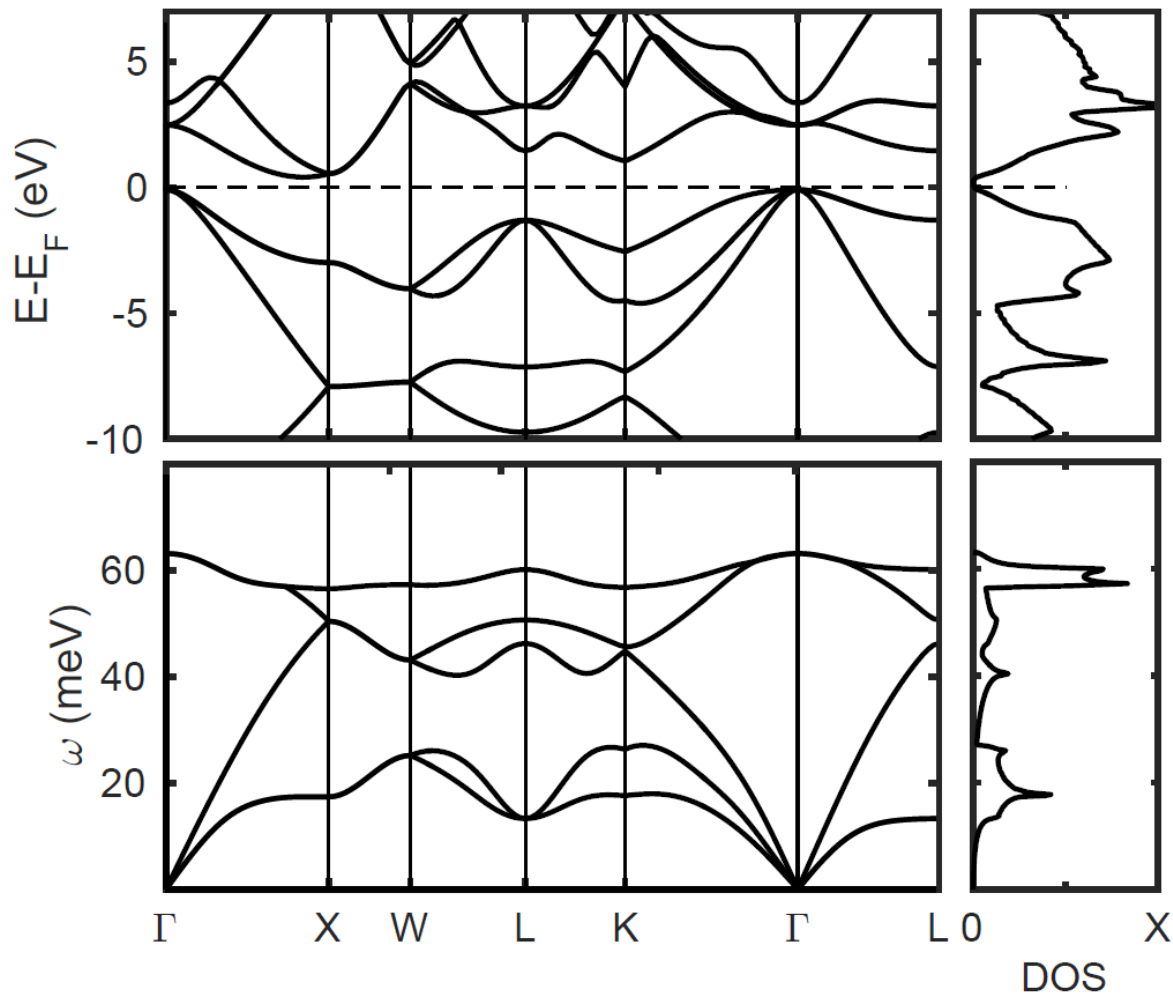
□ Spectral function

$$A_{n\mathbf{k}}(\omega, T) = \frac{1}{\pi} \frac{|\Sigma''_{n\mathbf{k}}(\omega, T)|}{|\omega - \varepsilon_{n\mathbf{k}} - \Sigma'_{n\mathbf{k}}(\omega, T)|^2 + |\Sigma''_{n\mathbf{k}}(\omega, T)|^2}$$

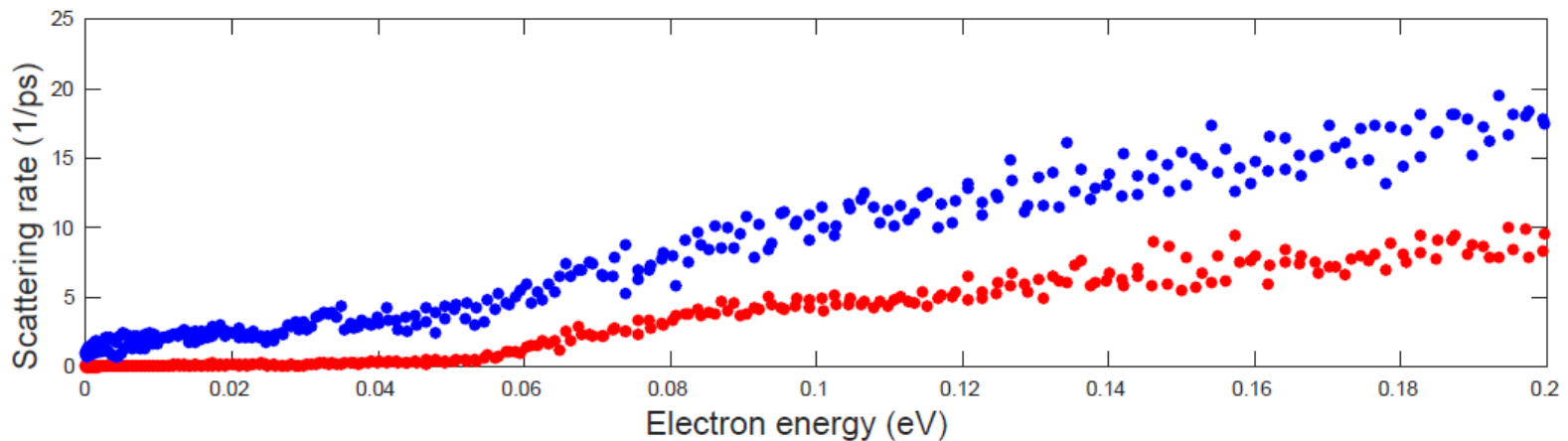
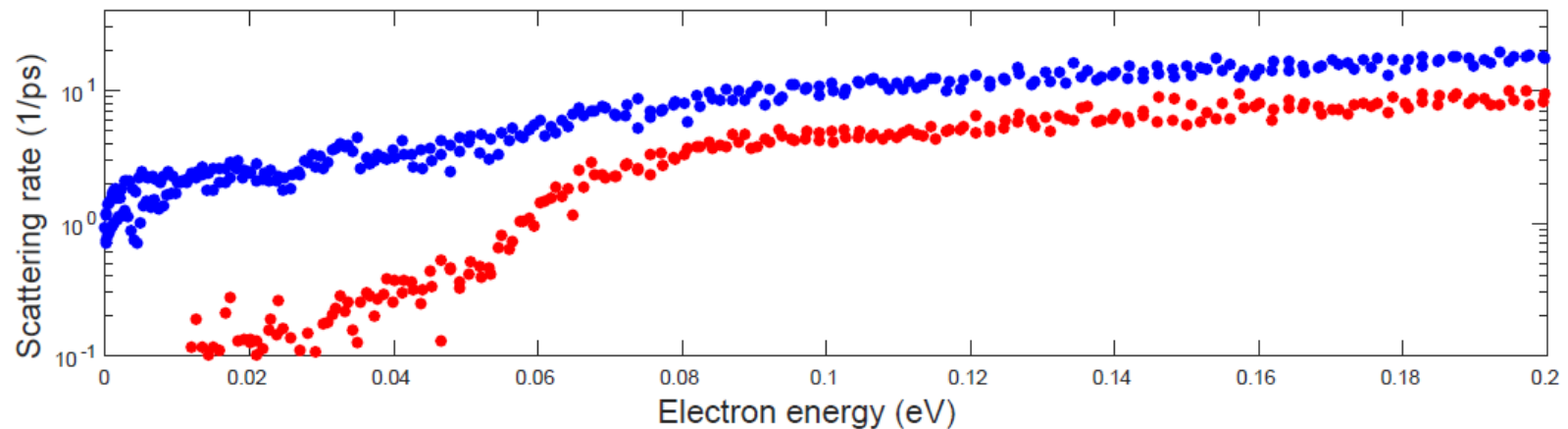
Test case: B-doped diamond



Test case: Si scattering rate



Test case: Si scattering rate from CBM

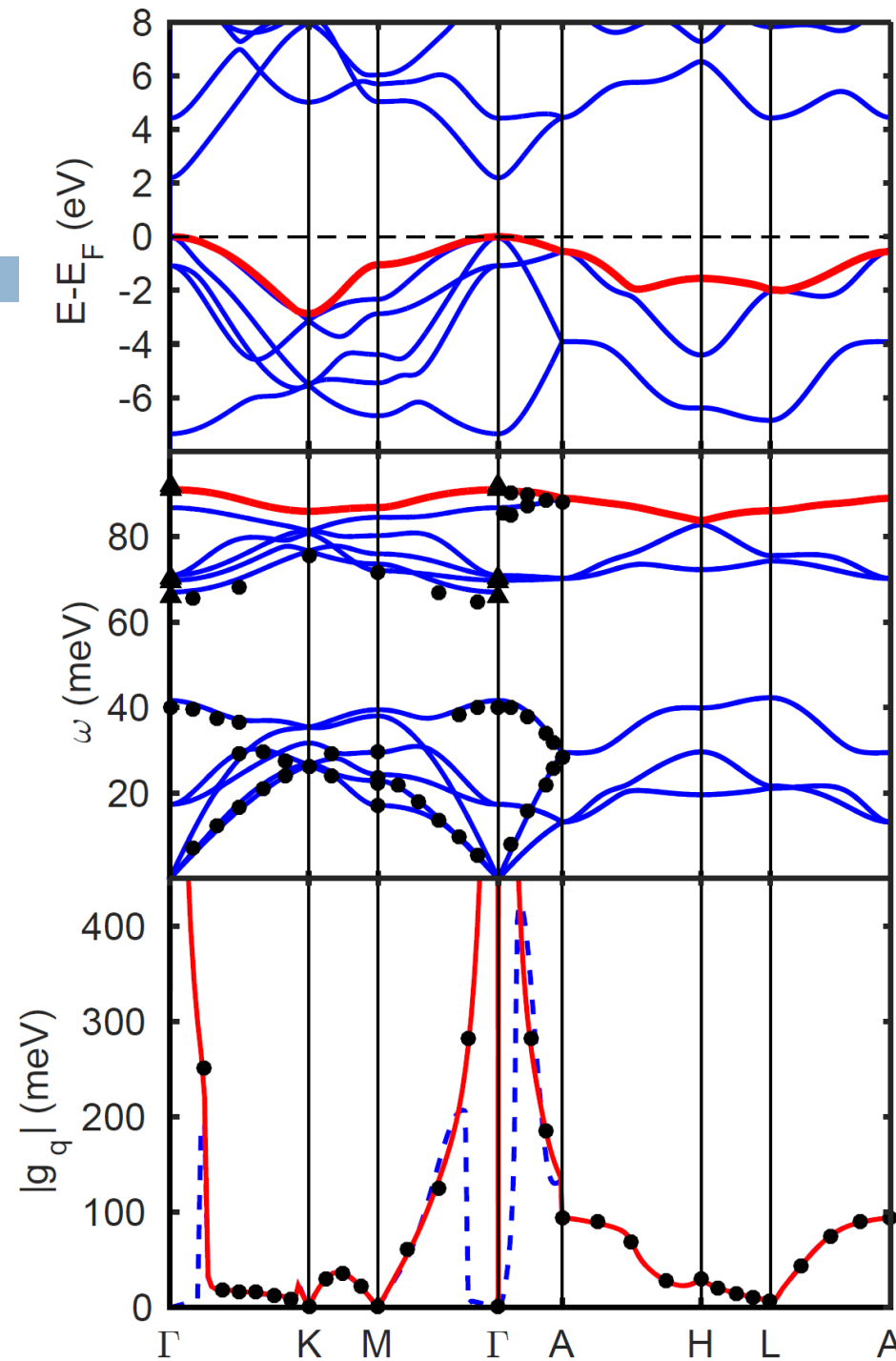


Test case: GaN

- The electron-phonon matrix elements:

$$g_{mn,\nu}(\mathbf{k}, \mathbf{q}) = \left(\frac{\hbar}{2m_0\omega_{\mathbf{q}\nu}} \right)^{1/2} \langle \psi_{m\mathbf{k}+\mathbf{q}} | \partial_{\mathbf{q}\nu} V | \psi_{n\mathbf{k}} \rangle$$

- Diverge as $\frac{1}{q}$ for $\mathbf{q} \rightarrow 0$
in polar materials



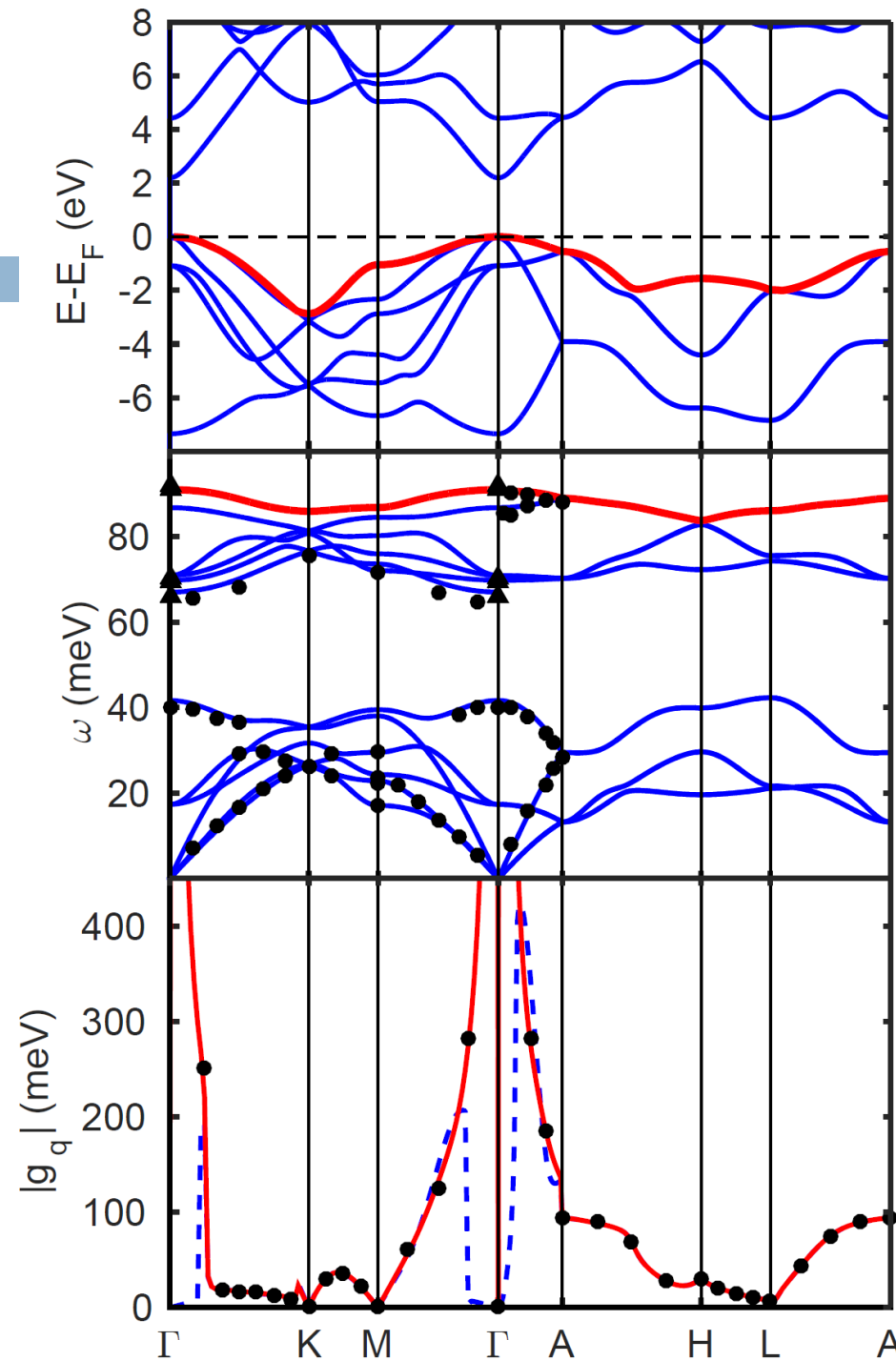
Test case: GaN

□ Correct Wannier interpolation:

$$g_{mn,v}(\mathbf{k}, \mathbf{q}) = g_{mn,v}^S(\mathbf{k}, \mathbf{q}) + g_{mn,v}^{\mathcal{L}}(\mathbf{k}, \mathbf{q})$$

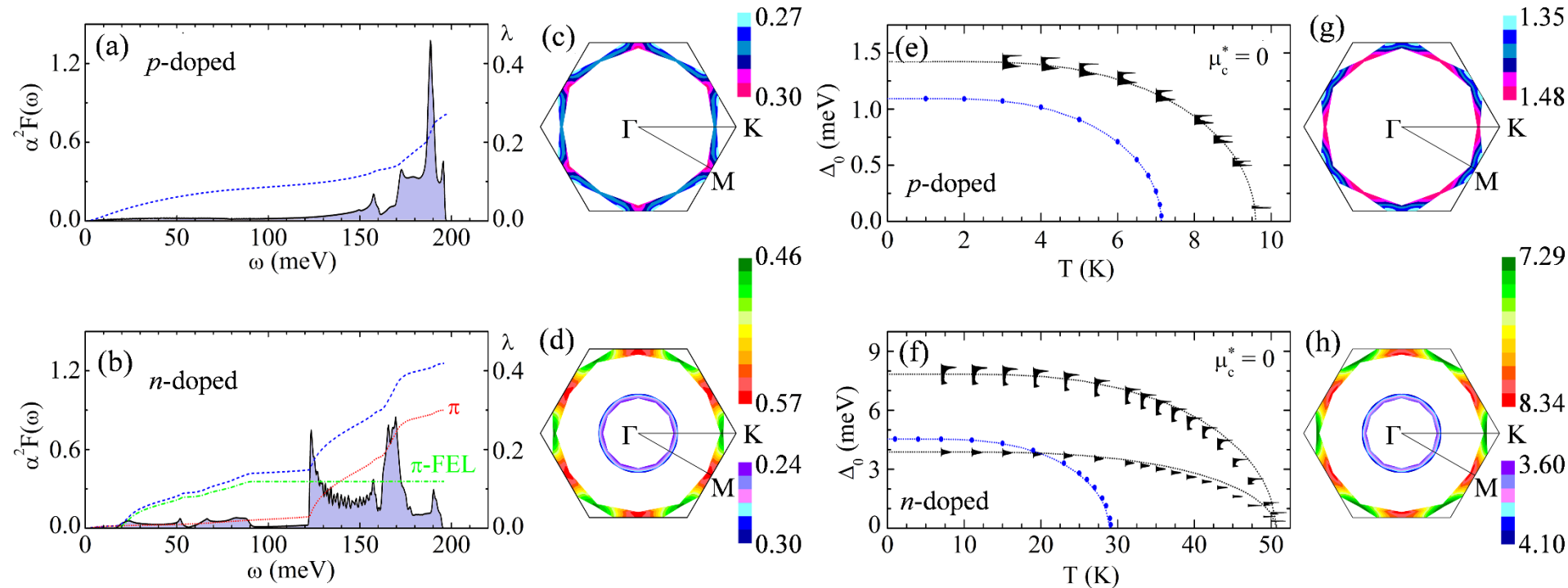
$$g_{mn,v}^{\mathcal{L}}(\mathbf{k}, \mathbf{q}) = i \sum_{\kappa} \left(\frac{\hbar}{2M_{\kappa}\omega_{\mathbf{q}\nu}} \right)^{1/2} \sum_{\mathbf{G} \neq -\mathbf{q}} \frac{(\mathbf{q} + \mathbf{G}) \cdot \mathbf{Z}_{\kappa}^* \cdot \mathbf{e}_{\kappa,\nu}(\mathbf{q})}{(\mathbf{q} + \mathbf{G}) \cdot \boldsymbol{\varepsilon} \cdot (\mathbf{q} + \mathbf{G})} \langle \psi_{m\mathbf{k}+\mathbf{q}} | e^{i(\mathbf{q}+\mathbf{G}) \cdot \mathbf{r}} | \psi_{n\mathbf{k}} \rangle$$

$$\langle \psi_{m\mathbf{k}+\mathbf{q}} | e^{i(\mathbf{q}+\mathbf{G}) \cdot \mathbf{r}} | \psi_{n\mathbf{k}} \rangle = [U_{\mathbf{k}+\mathbf{q}} U_{\mathbf{k}}^{\dagger}]_{mn}$$



Test case: doped-graphene

Superconducting gap & T_c



Conclusion & future developments

- Current version of EPW4
 - ▣ is fully compatible with QE and W90
 - ▣ is tested on different hardware & compilers with different libraries
 - ▣ has been optimized + new features implemented

- Future
 - ▣ G-vector parallelization
 - ▣ US (PAW) support
 - ▣ Expand the test farm: add mac hardware & power 8 (xlf)
 - ▣ Add new features related to transport (conductivity, resistivity etc...)

Acknowledgments



Carla Verdi



Roxana Margine