

# Phonon Unfolding: A program for unfolding phonon dispersions of materials

Fawei Zheng<sup>1,2,\*</sup> and Ping Zhang<sup>1,2,†</sup>

<sup>1</sup>*Institute of Applied Physics and Computational Mathematics, Beijing, People's Republic of China*

<sup>2</sup>*Beijing Computational Science Research Center, Beijing 100084, China*

(Dated: February 26, 2016)

We present **Phonon Unfolding**, a Fortran90 program for unfolding phonon dispersions. It unfolds phonon dispersions by using a generalized projection algorithm, which can be used to any kind of atomic systems in principle. Thus our present program provides a very useful tool for the phonon dispersion and vibration mode analysis of surface reconstructions, atomic point defects, alloys and glasses.

PACS numbers: 63.20.dk, 63.20.D-, 63.50.-x, 78.70.Nx

## PROGRAM SUMMARY

*Program Title:* Phonon Unfolding

*Catalogue identifier:*

*Programming language:* Fortran 90

*Computer:* any computer architecture

*RAM:* system dependent, about 10 MB

*Program obtainable from:*

*Number of processors used:* 1

*CPC Library subroutines used:* None

*Operating system:* Linux, Windows, Mac

*External routines/libraries:* LAPACK

*Keywords:* phonon dispersion unfolding, translational symmetry, neutron scattering

*Nature of problem:* The Brillouin zone of a supercell is smaller than that of a primary cell. It makes the supercell phonons more crowded. The crowded phonon dispersions are outright difficult, if not impossible, to be compared with experimental results directly. Besides, the intra-supercell translation symmetries are hidden in the supercell phonon dispersion calculations. In order to compare with experiments and catch the hidden symmetries, we have to unfold the supercell phonon dispersions into the corresponding primary-cell Brillouin zone.

*Solution method:* The phonon polarization vectors are projected to a group of plane waves. The unfolding weight is calculated from these plane wave components.

*Running time:* system dependent, from a few seconds to one hour

*Unusual features of the program:* Applicable to general systems without considering which kind of translational symmetry breaking. Simple and user-friendly input system. Great efficiency and high unfolding speed.

*References:*

F. Zheng, P. Zhang, "Phonon dispersion unfolding in the presence of heavy breaking of spatial translational symmetry", arXiv: 1602.06655.

## I. INTRODUCTION

Similar to the electronic Hamiltonian, the dynamical matrix for phonon is invariant under translation  $T(\mathbf{n}) = n_1\mathbf{a}_1 + n_2\mathbf{a}_2 + n_3\mathbf{a}_3$  in an ideal crystal. The  $\mathbf{a}_1 \sim \mathbf{a}_3$  are the lattice vectors of the crystal, and  $\mathbf{n}$  has three integer components  $(n_1, n_2, n_3)$ . The translational symmetry largely simplifies the crystal physics problems. And the application of Bloch's theorem to atomic vibration problem introduces the concept of phonon dispersions. The phonon dispersions are widely applied in condensed matter physics, especially in exploring the heat transport, atomic structure stabilities and thermal properties in crystal, and is connected to the neutron inelastic scattering experiments.

---

\*Electronic address: zheng\_fawei@iapcm.ac.cn

†Electronic address: zhang\_ping@iapcm.ac.cn

If the crystal contains translational symmetry breaking, for example the systems with point defects [1–3], interfacial reconstruction [4–6], complex spin configurations [7], and disorder [8–10], the natural choice in theoretical and computational study is to use the supercell method. However, the widely used supercell calculation scheme folds the phonon dispersions to supercell first Brillouin zone. The supercell first Brillouin zone is much smaller than that of the primitive cell. All the phonon dispersions become shorter and crowded in the limited reciprocal space. The shapes of phonon dispersions are destroyed, which makes it hard to analyze. The situation would be much more serious for the heavily folded phonon dispersions. The consequent phonon polarization vectors are supercell Bloch functions. Many of them have the same symmetries with primary-cell Bloch functions. It originates from the approximate primary-cell translational symmetry. Furthermore, the neutron inelastic scattering experimental results can not be referred to the supercell phonon dispersions and catch the hidden translational symmetry. Like the unfolded electron energy bands, we have to unfold the supercell phonon dispersions into the corresponding primitive cell first Brillouin zone.

Recently, the rapidly developed electron energy bands unfolding methods[9–22] stimulated the study of phonon unfolding methods[21–24]. In one of our previous papers[24], we proposed a generalized calculation scheme to unfold phonon dispersions, which can be applied to atomic systems with any kind of translational symmetry broken. Let's denote the phonon polarization vector as,

$$\psi_{\vec{q}}^s(\vec{r}_i) = \phi_{\vec{q}}^s(\vec{r}_i)e^{i\vec{q}\cdot\vec{r}_i}$$

$$(i = 1, 2, \dots, N, s = 1, 2, 3).$$

Here, the band index is omitted. The integer  $s = 1, 2, 3$  corresponds to  $x, y,$  and  $z$  directions respectively. The  $\vec{q}$  is the wave number in the supercell first Brillouin zone, and  $\vec{r}_i$  denotes the  $i$ -th atom position. The integer  $N$  is the supercell total atom number. Suppose the supercell contains  $n$  primitive cells. Then, the primitive cell first Brillouin zone contains  $n$  Brillouin zones of supercell. The projection operator for each supercell Brillouin zone is noted as  $\hat{P}_b$  ( $b = 1..n$ ). As an approximation, we use the plane wave basis to construct the projection operator as,

$$\hat{P}_b = \sum_{j,s} |w_{j,s,b}\rangle \langle w_{j,s,b}|$$

In which,

$$|w_{j,s,b}\rangle = \frac{e^{i(\vec{Q}_b + \vec{G}_j)\cdot\vec{r}_i}}{\sqrt{N}} \delta_{s,s'}$$

Where the  $\vec{Q}_b$  is the reciprocal lattice point of supercell. The  $\vec{G}_j$  is the reciprocal lattice point of primitive cell, i.e. it is the wavenumber of plane wave basis. We symmetrically chose finite number of  $\vec{G}_j$  around  $(0, 0, 0)$  point. Then the unfolding weight can be written as:

$$|c_{\vec{q}+\vec{Q}_b}|^2 = \langle \phi_{\vec{q}} | \hat{P}_b | \phi_{\vec{q}} \rangle$$

$$= \sum_{j,s} |B_{j,s}^{\vec{q}+\vec{Q}_b}|^2$$

$$B_{j,s}^{\vec{q}+\vec{Q}_b} = \langle w_{j,s,b} | \phi_{\vec{q}} \rangle$$

$$= \sum_i \frac{e^{-i(\vec{Q}_b + \vec{G}_j)\cdot\vec{r}_i}}{\sqrt{N}} \phi_{\vec{q}}^s(\vec{r}_i).$$

The present code **PhononUnfolding** is based on this plane wave projection scheme. It is applicable to any kind of symmetry breaking systems, even if the system is heavily symmetry broken, when the atom correspondence is not available. Presently, the code contains interface to Quantum Espresso package.

## II. BRIEF DESCRIPTION OF THE CODE

After decompressing the *PhononUnfolding.tar.gz* file, one gets a folder named as *PhononUnfolding*. There are *Readme* file and four sub-folders. They are *src*, *system*, *doc* and *examples*. Sub-folder *src* contains two Fortran90 source files, *src* contains *makefile\_windows* and *makefile\_linux* which can be used for compiling the code in Windows and Linux operating systems respectively, *doc* folder contains a description file of the present code.

In this code, the control parameters are read from the input.dat file, which will be described in detail in the next section. After we obtain the force constants parameters, there are three steps to get the unfolded phonon dispersions.

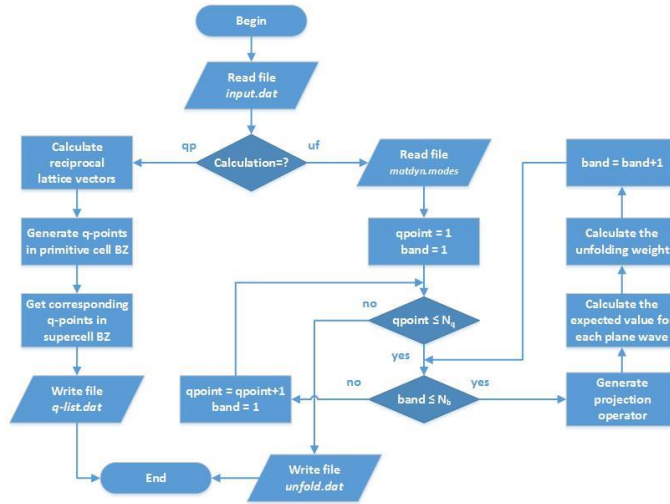


FIG. 1: Flow chart of phonon dispersions unfolding process in **Phonon Unfolding**. The BZ is the abbreviate of Brillouin zone.

First of all, we have to produce the q-point list in the first Brillouin zone of supercell. The parameter *calculation* is set to *qp*. Then the program reads data block *begin primary cell qpoint ~ end primary cell qpoint*. It generates q-point list along the high symmetry lines in the primitive cell Brillouin zone. Then, the program read data block *begin primitive cell vectors ~ end primitive cell vectors* and *begin super cell vectors ~ end super cell vectors*. From the supercell and primitive cell vectors, the program calculate the q-point correspondence between them, and calculate the q-point list in supercell Brillouin zone. The q-point list is written in file *q-list.dat*. If parameter *write\_q\_correspondence = true*, then file *Q-points.dat* will be produced, which contains more details of the q-point list.

Then, we use the q-point list to calculate the supercell phonon polarization vectors. At the last step, we set *calculation = uf*. The *PhononUnfolding* code reads the polarization vectors produced by *matdyn.x* in Quantum Espresso package. Interfaces to other first-principle and classical molecular dynamics codes will be produced in the near future. The *PhononUnfolding* program read parameters *max\_qx*, *max\_qy* and *max\_qz*, and gets the plane wave basis. The basis functions are used to get the projection operator  $\hat{P}_b$ . From which, we finally obtain the phonon unfolding weight. The calculation results are stored in file *unfold.dat*, which can be plotted by using Origin or Gnuplot. The detailed flow chart of unfolding process is shown in Fig. 1.

### III. THE INPUT.DAT FILE

The target to be performed and the information of the system can be described by the keywords in input.dat file. The ordering of the keywords is not significant. Case is ignored, so that **calculation** is the same as **Calculation**. Characters after ! or # are treated as comments. Most keywords have default values unless they are given in input.dat. The keywords are described as follows:

- **calculation** = uf | qp  
*Default value* : uf  
*Value type* : string of characters

The keyword ‘**calculation**’ describes the task to be performed. The value of ‘**calculation**’ has two options at the present time; they are:

- uf : phonon dispersion unfolding.
- qp : q-point list generation.

- **max\_qx,max\_qy,max\_qz** = 0, 1, 2 ...  
*Default value* : 2  
*Value type* : integer number

**max\_qx**, **max\_qy**, and **max\_qz** define the plane wave groups to be used in projection calculation. The wave numbers are:  $-\mathbf{max\_qx} < n_x < \mathbf{max\_qx}$ ,  $-\mathbf{max\_qy} < n_y < \mathbf{max\_qy}$  and  $-\mathbf{max\_qz} < n_z < \mathbf{max\_qz}$ . For

low dimensional systems, the non-periodic dimension value can be simple set to zero. For example, **max\_qz** = 0 in graphene system in x-y plane.

- **wtclean** = 0 ~ 1

*Default value* : 0.01

*Value type* : real number

The result data point with unfolding weight lower than ‘**wtclean**’ will not be written to the output file. Then the output file would be smaller and is easier to handle.

- **write\_q\_correspondence**= t | true | f | false

*Default value* : false

*Value type* : logic

The value is case insensitive. If ‘**write\_q\_correspondence**’ = true or t, then a *Q-points.dat* file is produced, which contains all the q point list in primitive cell Brillouin zone and the corresponding q point list in super cell Brillouin zone. There are seven columns in *Q-points.dat* file. The first column is the high symmetry path length, the next three columns are q point list in first Brillouin zone of primitive cell, the last three columns are the corresponding q point list in the first Brillouin zone of supercell.

- **begin primary cell qpoint** and **end primary cell qpoint**

The data block between ‘**begin primary cell qpoint**’ and ‘**end primary cell qpoint**’ defines the high-symmetry lines in the first Brillouin zone of the primitive cell. There are  $3n$  lines in the data block. Each three lines define one high-symmetry q-point line. The first line is an integer, which is larger than 1. It shows the number of q-points along the high-symmetry q-point line. Both the second and the third lines have three real numbers, which show the starting and end points of the high-symmetry q-points line in direct form.

- **begin primitive cell vectors** and **end primitive cell vectors**

The data block between ‘**begin primitive cell vectors**’ and ‘**end primitive cell vectors**’ have three lines. They define the three cell vectors of primitive cell with the unit of angstrom.

- **begin super cell vectors** and **end super cell vectors**

The data block between ‘**begin super cell vectors**’ and ‘**end super cell vectors**’ have three lines. They define the three cell vectors of super cell with the unit of angstrom.

- **begin super cell atom positions** and **end super cell atom positions**

The data block between ‘**begin super cell atom positions**’ and ‘**end super cell atom positions**’ show the atom positions of supercell with the unit of angstrom. Each line have three real numbers which is the Cartesian position of the corresponding atom. The order of the atoms in this data block should be the same with that of phonon polarization vector file *matdyn.modes*.

## IV. EXAMPLES

In the following context we illustrate the capabilities of **Phonon Unfolding** by describing two systems: (i) graphene, a two dimensional system without translational symmetry breaking; (ii) diamond with a carbon vacancy, a three dimensional system with translational symmetry breaking; The unfolding process is trivial in graphene. The calculation results should coincide with the phonon dispersions from the primitive-cell calculations. The example here shows the validity of the present code. The translational symmetries in diamond with carbon vacancy are broken. Then, the unfolding process is nontrivial.

### A. Graphene

Since the success of fabrication single layer graphene[26], this material has been drawing much attentions[32–37] due to its peculiar properties[27–31]. Similar to graphite, *h*-BN, and carbon nanotubes, graphene has two-dimensional honeycomb structure. It is the building block of fullerene, carbon nanotubes, graphene nanoribbons, and graphite.

As an example, we consider phonon dispersion unfolding of a freestanding graphene. The structure relaxation and electronic structure calculations are performed by using DFT [38, 39] with norm-conserving carbon pseudopotential

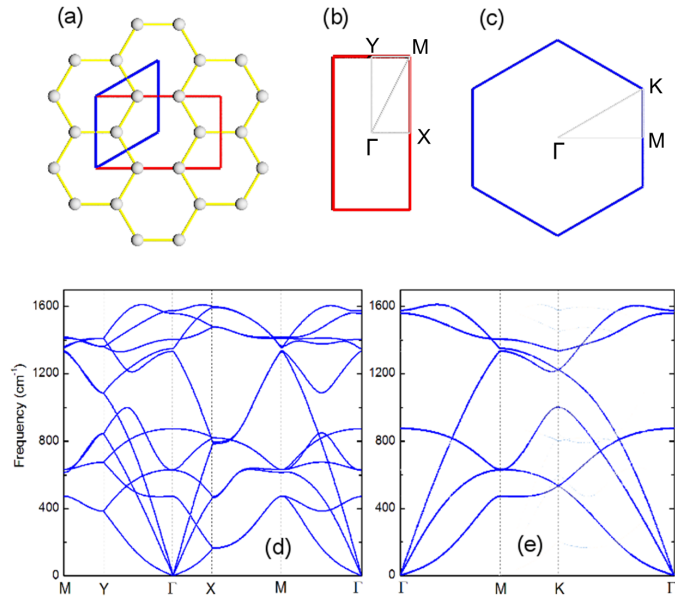


FIG. 2: (a) Atomic structure of graphene with its supercell (red rectangle) and primary cell (blue parallelogram). Panels (b) and (c) show the first Brillouin zones of the supercell and the primary cell, respectively, with high symmetry points and lines. Panel (d) shows the phonon dispersions of the supercell. Panel (e) shows the phonon dispersions unfolded into the primary-cell Brillouin zone.

[40, 41]. The exchange correlation potential is described by the generalized gradient approximation (GGA) of Perdew-Burke-Ernzerhof (PBE) type [42]. The kinetic energy cutoff for wavefunction is chosen to be 120 Ry, which is converged in our test. We use a rectangle supercell in our DFT calculations. Each supercell contains four carbon atoms as shown in Fig. 2. The graphene layers are separated by a vacuum of 20 Å in order to reduce the interactions between the nearest layers. The system is relaxed until the force on each atom is smaller than 0.005 eV/Å. BFGS quasi-newton algorithm is used in the structure relaxation. In the self-consistent ground state calculations,  $9 \times 16 \times 1$  Monkhorst-Pack K-points setting is used in the reciprocal space integration. After obtaining the self-consistent ground state, we performed the density functional perturbation theory (DFPT) calculations with a uniform  $3 \times 5 \times 1$  grid of q-points setting. Then perform dynamical matrix Fourier transformations to get force constants, and calculated the phonon dispersions in graphene supercell. All the DFT and DFPT calculations are performed by using Quantum Espresso package [43].

The atomic structure of graphene is shown in Fig. 2(a). In which, the red rectangle and blue parallelogram are the supercell and primitive cell respectively. The supercell and primitive cell contain 4 and 2 carbon atoms respectively. Their first Brillouin zone and the high symmetry lines used in calculations are shown in Fig. 2(b,c). The calculated phonon dispersions in graphene supercell are shown in Fig. 2(d). Then we calculate the q points list in primitive cell Brillouin zone high-symmetry lines, and the corresponding q points list in supercell Brillouin zone. This can be done by setting *calculation = qp* in *input.dat* file. After performing *PhononUnfolding.exe*, the q points list is stored in the *q-list.dat* file. We copy the list to the input file of *matdyn.x* in Quantum Espresso package. And calculate the phonon dispersions and polarization vectors (stored in *matdyn.modes* file) in supercell. After that, we set *calculation = uf* in *input.dat*, and run *PhononUnfolding.exe* again. The calculation results are written in *unfold.dat* file. The plotted unfolded phonon dispersions are shown in Fig. 2(e), which is the same with phonon dispersions in primitive cell graphene[25].

## B. Diamond with Vacancy

In order to test the code in three-dimensional systems, we further calculate the phonon dispersions of a diamond supercell with a carbon vacancy, and unfold it to the primary-cell Brillouin zone. The DFT calculations are carried out to relax the atomic structure. The inner electrons of carbon atoms are described by norm-conserving pseudopotentials [40, 41]. The exchange correlation potential is described by the GGA of PBE-type [42]. The kinetic energy cutoff for wavefunction is chosen to be 120 Ry, which is converged in our test. We use a cubic supercell in our DFT calculations as shown in Fig. 3(a). Each supercell contains seven carbon atoms and one carbon vacancy. The carbon atoms and

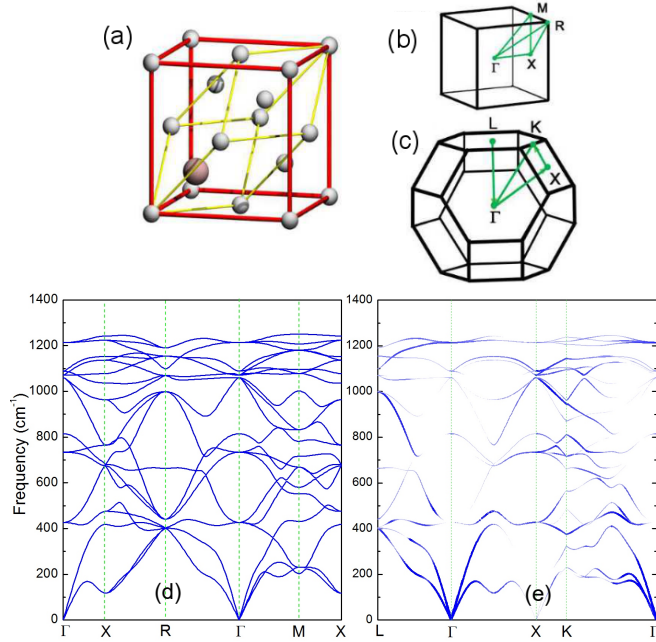


FIG. 3: (a) Atomic structure of diamond with one vacancy. The supercell and primitive cell are shown by a red and yellow cages. The C atoms and vacancy are shown by gray and brown balls, respectively. Panels (b) and (c) show the first Brillouin zones of supercell and primitive cell with high symmetry lines. Panels (d) shows the phonon dispersions of the supercell, and panel (e) shows the phonon dispersions unfolded into the primitive cell Brillouin zone.

vacancy are shown by gray and brown balls respectively. The primary cell is also shown in Fig. 3(a) by a yellow cage. The first Brillouin zone and high symmetry lines used in our calculations of supercell and primitive cell are shown in Fig. 3(b) and (c) respectively. The relaxed supercell lattice parameter is 3.57 Å. The system is relaxed until the force on each atom is smaller than 0.005 eV/Å. BFGS quasi-newton algorithm is used in the structure relaxation. In the self-consistent ground state calculation, a  $13 \times 13 \times 13$  Monkhorst-Pack K-points setting is used in the reciprocal space integration. After obtaining the self-consistent ground state, we performed the DFPT calculations with  $4 \times 4 \times 4$  grid of q-point setting. The force constants were obtained by Fourier transport of dynamic matrix. The supercell phonon dispersions calculated by using the force constants are shown in Fig. 3(d). They are complex compare to the primitive cell phonon dispersions due to the additional atoms and an extra vacancy.

Then we continue to calculate the unfolded phonon dispersions. Similar to the case of graphene, we first generate the q-point list by setting *calculation = qp* in *input.dat* file. After running *PhononUnfolding.exe*, the q points list is stored in the *q-list.dat* file. We copy the q-list to the input file of *matdyn.x*. Then we calculate the phonon dispersions and polarization vectors in supercell. The phonon polarization vectors are stored in *matdyn.modes* file. After that, we set *calculation = uf* in *input.dat*, and run *PhononUnfolding.exe* again. The program loads the phonon polarization vectors and produces the phonon dispersion unfolding weight. The calculation results are written in *unfold.dat* file. The plotted unfolded phonon dispersions are shown in Fig. 3(e). The unfolded energy bands of the doped system have broken points and darkness in a variety, which originate from the breaking of translational symmetry.

## V. CONCLUSION

In this communication we introduced **Phonon Unfolding**, a computer code for unfolding phonon dispersions by using. **Phonon Unfolding** enables accurate and efficient calculations of the phonon dispersions. The executable versions of **Phonon Unfolding** for Windows and Linux operation systems are distributed by email. Plans are in place to extend **Phonon Unfolding** in order to unfold phonon dispersions calculated by Phon[44], Phonon[45], Phonopy[46], ABINIT[47], Siesta[48], and LAMMPS[49].

## VI. ACKNOWLEDGMENTS

The research leading to these results has received funding from Natural Science Foundation of China (NSFC) under Grant Nos.11474030, U1530258 and the joint grant of NSFC and NSCC-GZ.

- 
- [1] E. van Heumen, J. Vuorinen, K. Koepernik, *et al.* Phys. Rev. Lett. **106** (2011) 027002.
- [2] S. Konbu, K. Nakamura, H. Ikeda, and R. Arita, Solid State. Commun. **152** (2012) 728.
- [3] F. Zheng, L.-L. Wang, Q.-K. Xue, and P. Zhang, Phys. Rev. B **93** (2016) 075428.
- [4] S. Kim, J. Ihm, H. J. Choi, and Y.-W. Son, Phys. Rev. Lett. **100** (2008) 176802.
- [5] Y. Qi, S. H. Rhim, G. F. Sun, M. Weinert, and L. Li, Phys. Rev. Lett. **105**, (2010) 085502.
- [6] S. Cahangirov, M. Audiffred, P. Tang, A. Iacomino, W. Duan, G. Merino, and A. Rubio, Phys. Rev. B **88** (2013) 035432.
- [7] K. Liu, Z.-Y. Lu, and T. Xiang, Phys. Rev. B **85** (2012) 235123.
- [8] M. W. Haverkort, L. S. Elfmov, and G. A. Sawatzky, arXiv:1109.4036.
- [9] V. Popescu, and A. Zunger, Phys. Rev. B **85** (2012) 085201.
- [10] V. Popescu, and A. Zunger, Phys. Rev. Lett. **104** (2010) 236403.
- [11] T. B., Boykin, N. Kharche, G. Klimeck, and M. Korkusinski, J. Phys.: Condens. Matter **19** (2007) 036203
- [12] T. B., Boykin, and G. Klimeck, Phys. Rev. B **71** (2005) 115215.
- [13] W. Ku, T. Berlijn, and C.-C. Lee, Phys. Rev. Lett. **104** (2010) 216401.
- [14] S. Konbu, K. Nakamura, H. Ikeda, and R. Arita, J. Phys. Soc. Japan **80**, (2011) 123701.
- [15] C.-C. Lee, Y. Yamada-Takamura, and T. Ozaki, J. Phys.: Condens. Matter **25** (2013) 345501.
- [16] F. Zheng, P. Zhang, and W. Duan, Comput. Phys. Commun. **189** (2015) 213.
- [17] O. Rubel, A. Bokhanchuk, S. Ahmed, and E. Assmann, Phys. Rev. B **90**, (2014) 115202.
- [18] I. Deretzis, G. Calogero, G. Angilella, and A. La Magna, EPL (Europhysics Letters) **107**, (2014) 27006.
- [19] P. Brommer and D. Quigley, J. Phys.: Condens. Matter **26**, (2014) 485501.
- [20] M. Farjam, arXiv: 1504.04937.
- [21] H. Huang, F. Zheng, P. Zhang, J. Wu, B.-L. Gu and W. Duan, New J. Phys. **16** (2014) 033034.
- [22] P. B. Allen, T. Berlijn, D. A. Casavant, and J. M. Soler, Phys. Rev. B **87** (2013) 085332.
- [23] T. B. Boykin, A. Ajoy, H. Ilatikhameh, M. Povolotskyi, and G. Klimeck, Phys. Rev. B **90**, (2014) 205214.
- [24] F. Zheng, and P. Zhang, arXiv: 1602.06655.
- [25] R. Saito, G. Dresselhaus, and M. Dresselhaus, *Physical Properties of Carbon Nanotubes*, Imperial College Press, London, 1998.
- [26] K. S. Novoselov, A. K. Geim, S. V. Morozov, D. Jiang, Y. Zhang, S. V. Dubonos, I. V. Grigorieva, and A. A. Firsov, Science **306** (2004) 666.
- [27] A. H. Castro Neto, F. Guinea, N. M. R. Peres, K. S. Novoselov, and A. K. Geim, Rev. Mod. Phys. **81**, (2009) 109.
- [28] A. K. Geim, and K. S. Novoselov, Nat. Mater. **6**, (2007) 183.
- [29] K. S. Novoselov, D. Jiang, F. Schedin, T. J. Booth, V. V. Khotkevich, S. V. Morozov, and A. K. Geim, Proc. Natl. Acad. Sci. U.S.A. **102** (2005) 10451.
- [30] K. S. Novoselov, A. K. Geim, S. V. Morozov, D. Jiang, M. I. Katsnelson, I. V. Grigorieva, S. V. Dubonos, and A. A. Firsov, Nature **438** (2005) 197.
- [31] Y. Zhang, Y.-W. Tan, H. L. Stormer, and P. Kim, Nature **438** (2005) 201.
- [32] Q. Yan, B. Huang, J. Yu, F. Zheng, J. Zang, J. Wu, B.-L. Gu, F. Liu, and W. Duan, Nano Lett. **7** (2007) 1469.
- [33] N. Mohanty, and V. Berry, Nano Lett. **8** (2008) 4469.
- [34] L. Liao, Y.-C. Lin, M. Bao, R. Cheng, J. Bai, Y. Liu, Y. Qu, K. L. Wang, Y. Huang, and X. Duan, Nat. **467** (2010) 305.
- [35] F. Xia, T. Mueller, R. Golizadeh-Mojarad, M. Freitag, Y. Lin, J. Tsang, V. Perebeinos, and P. Avouris, Nano Lett. **9** (2009) 1039.
- [36] R. Raccichini, A. Varzi, S. Passerini, and B. Scrosati, Nat. Mater. **14** (2015) 271.
- [37] S. J. Heerema, and C. Dekker, Nat. Nanotechnol. **11** (2015) 127.
- [38] P. Hohenberg, and W. Kohn, Phys. Rev. **136** (1964) B864.
- [39] W. Kohn, , and L. J. Sham, Phys. Rev. **140** (1965) A1133.
- [40] D. R. Hamann, M. Schlüter, and C. Chiang, Phys. Rev. Lett. **43** (1979) 1494.
- [41] D. R. Hamann, Phys. Rev. B **40** (1989) 2980.
- [42] J. P. Perdew, K. Burke, and M. Ernzerhof, Phys. Rev. Lett. **77** (1996) 3865.
- [43] P. Giannozzi, S. Baroni, N. Bonini, et al., J. Phys.: Condens. Matter **21** (2009).
- [44] D. Alfè, Comput. Phys. Commun. **180** (2009) 2622.
- [45] K. Parlinski, AIP Conf. Proc. **479** (1999) 121.
- [46] A. Togo., (2009). Program available at <http://phonopy.sourceforge.net>.
- [47] X. Gonze, Comput. Phys. Commun. **180** (2009) 2582.
- [48] J. Soler, E. Artacho, J. D. Gale, A. García, J. Junquera, P. Ordejón, and D. Sánchez-Portal, J. Phys.: Condens. Matter **14** (2002) 2745.
- [49] S. Plimpton, P. Crozier, and A. Thompson, (2007). Program available at <http://http://lammps.sandia.gov/>.