

# INVESTIGATING GRAPHENE-LIKE MOIRÉ TOPOLOGICAL SUPERCONDUCTORS

**Rochish Manda**

M.Sc. Nanotechnology, Kungliga Tekniska högskolan

## ABSTRACT

The primal objective concerned in this work is studying moiré analogues of Graphene in terms of their electronic band structure topology. In light of this objective and other considerations entailed in the study, single layer Silicene was chosen as a material of interest to study its topology through its berry curvature signature. The degree of bulk-boundary correspondence was established through the berry curvature integral computation for Silicene and Graphene. It is expected that as a constituent atom goes down a group in the periodic table for a crystal system of choice, the degree of bulk-boundary correspondence increases proportionally due to a higher degree of electron-correlations in the respective Brillouin zones. And this study infers it for Group IV through Silicene and Graphene.

**Keywords**— Silicene, Topology, Berry Phase

## 1 INTRODUCTION

Ever since topological materials were discovered and extensively studied upon in recent years, their novel near-zero energy dissipation realisations such as in FET transistors, superconducting qubits and thermoelectrics are becoming a reality. This is only possible due to their unique bulk-boundary correspondence leading to the edges or surfaces of these materials being topologically protected from the external environment. Magic-angle twisted multilayer moiré materials have been shown to be one of amongst many promising classes of topological materials exhibiting superconducting phases of matter and approaching viable operations at higher relative temperatures above absolute zero; graphene being such a topological example due to an inherent collective electron excitation resulting from an inherent relativistic electron motion.

High temperature (near 298K) superconductivity remains something that is necessarily desirable yet unachievable with current known catalogue of materials. The main aim of this research project is then to identify such potential graphene-like topological moiré materials which can operate at room temperatures, by performing numerical calculations on their topology and band structures. Key features that were kept in consideration while identifying such candidate moiré material include – ease of fabrication (cost, sustainably-sourced and abundance-wise) and the degree of bulk-boundary correspondence.

Considering time-constraints for this project and aforementioned features for the candidate material, a moiré monolayer analogue of Graphene – **Silicene** was chosen as a material of interest to analyse and compare its electronic band structure with that of Graphene, and subsequently its topology through the respective Berry Phase signature. Silicene is known to be slightly corrugated ( $\beta$ -type) honeycomb structure (Seymour Cahangirov & Guy Le Lay, 2017), but this buckling is not known to affect its feature of linearly cross bands near the Fermi level (Miró et al., 2014). Neither is the change in hybridisation (from  $sp^2$  in C to isoelectronic  $sp^3$  in Si) known to affect the linear dispersion around the Dirac points of Silicene (Miró et al., 2014). Hence, Silicene deemed fit for potential candidate after Graphene, given that silicon is already well integrated in the electronics industry.

## 2 METHODOLOGY

Numerical calculations were performed on a tight-binding model of Silicene unit cell, which is similar to that of Graphene (hexagonal lattice P63/mmc (Miró et al., 2014)). Hence, the only parametric difference between Silicene and Graphene to be implemented in this tight-binding calculation would be the lattice constant, or alternatively the on-site energy.

Numerical calculation for band structure and berry phase integral was carried out using the Python package for tight-binding approximation **PythTB**, the implementation of which is shown in the appendix section [4] for the respective instances. As discussed in the above sections, the only difference between PythTB implementations of Silicene and Graphene would be the on-site energy, which is the hopping parameter 't'. The values of 't' for Silicene ( $V_{pp\pi} = -1.12\text{eV}$ ) and Graphene ( $V_{pp\pi} = -3.033\text{eV}$ ) were adapted from Hattori et al. (2017), since the on-site energies and hopping parameters were already calculated using ab-initio methods in Hattori et al. (2017).

### 3 RESULTS AND DISCUSSION

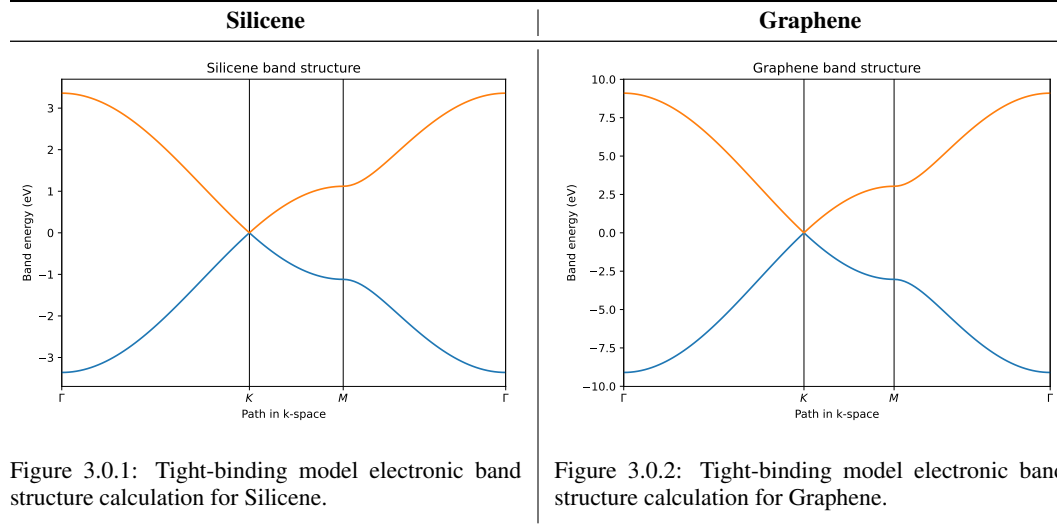


Figure 3.0.1: Tight-binding model electronic band structure calculation for Silicene.

Figure 3.0.2: Tight-binding model electronic band structure calculation for Graphene.

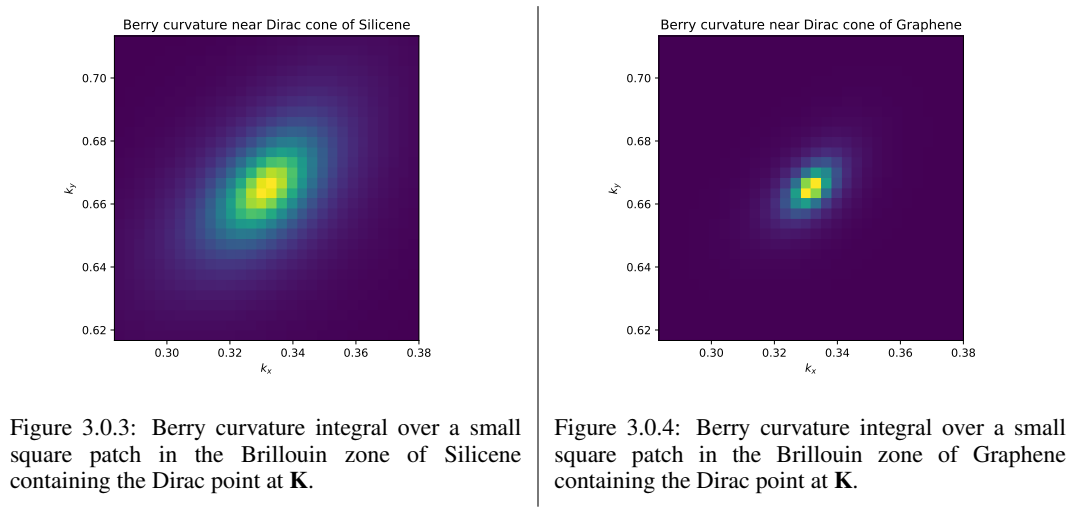


Figure 3.0.3: Berry curvature integral over a small square patch in the Brillouin zone of Silicene containing the Dirac point at **K**.

Figure 3.0.4: Berry curvature integral over a small square patch in the Brillouin zone of Graphene containing the Dirac point at **K**.

Table 3.0.1: Compilation of Electronic band structure and Berry flux calculations in PythTB for Silicene and Graphene, respectively.

From the results in Table 3.0.1, the following observations have been made –

1. Both Silicene and Graphene in Figure 3.0.1, 3.0.2 have Dirac points at **K** of their Brillouin zone, but Silicene has lower band gap than Graphene at all other points in the k-space.
2. Silicene has higher berry curvature integral than Graphene (Figure 3.0.3, 3.0.4) near the Dirac cone, implying that Silicene has higher Chern number than Graphene – which may also imply higher bulk-boundary correspondence in Silicene than Graphene.

From the observation on berry curvature integrals for Silicene and Graphene near the Dirac cone, it can be inferred that in a given crystal system, as the constituent's atomic number increases down its elemental group,

the degree of electronic correlations increase giving rise to a higher berry flux. And since a higher berry flux may imply a higher bulk-boundary correspondence, then in order to achieve high temperature superconductivity – which is very much dependent on the degree of bulk-boundary correspondence; crystal systems of higher atomic number might be of good interest.

The temperature at which monolayer Silicene becomes superconductive is yet to be measured experimentally, but just from a berry flux magnitude perspective in Figures 3.0.3 and 3.0.4, it should be theoretically higher than Graphene. While a lower band gap in Silicene at other than **K** points might make it more prone to ‘Boltzmann tyranny’ at room temperature operations (26 meV) than Graphene, the higher berry flux in Silicene might offer better topological protection from the external environment in the superconducting regime.

## 4 CONCLUSIONS AND PERSPECTIVES

This work on Silicene was built upon the existing framework of tight-binding graphene model, with the addition of Berry flux computation for a qualitative estimation of bulk-boundary correspondence - a key ingredient for near-zero energy dissipation realisations. And with the obtained results for Silicene – of higher bulk-boundary correspondence and lower band gap than Graphene (except at K-point), it seems as a potential successor to Graphene for the many topological applications yet to come, if it is viable to be more sustainably synthesised than Graphene.

## ACKNOWLEDGMENTS

The author would like to thank the developers of PythTB (Sinisa Coh and David Vanderbilt) for making their code available on a free and open-source basis.

## REFERENCES

- Ayami Hattori, Sho Tanaya, Keiji Yada, Masaaki Araidai, Masatoshi Sato, Yasuhiro Hatsugai, Kenji Shiraishi, and Yukio Tanaka. Edge states of hydrogen terminated monolayer materials: silicene, germanene and stanene ribbons. 29(11):115302, feb 2017. doi: 10.1088/1361-648x/aa57e0.
- Pere Miró, Martha Audiffred, and Thomas Heine. An atlas of two-dimensional materials. *Chemical Society reviews*, 43, 05 2014. doi: 10.1039/c4cs00102h.
- Hasan Sahin Seymur Cahangirov and Angel Rubio Guy Le Lay. *Introduction to the Physics of Silicene and other 2D Materials*, volume 930. Springer International Publishing, 2017.

## A APPENDIX 1

```
from __future__ import print_function
from pythtb import * # import TB model class
import numpy as np
import matplotlib.pyplot as plt

# define lattice vectors
lat=[[1.0,0.0],[0.5,np.sqrt(3.0)/2.0]]
# define coordinates of orbitals
orb=[[1./3.,1./3.],[2./3.,2./3.]]

# make two dimensional tight-binding graphene model
my_model=tb_model(2,2,lat,orb)

# set model parameters
delta=0.0
t=-1.12

# set on-site energies
my_model.set_onsite([-delta,delta])
# set hoppings (one for each connected pair of orbitals)
# (amplitude, i, j, [lattice vector to cell containing j])
my_model.set_hop(t, 0, 1, [ 0, 0])
```

```
my_model.set_hop(t, 1, 0, [ 1, 0])
my_model.set_hop(t, 1, 0, [ 0, 1])

# print tight-binding model
my_model.display()

# generate list of k-points following a segmented path in the BZ
# list of nodes (high-symmetry points) that will be connected
path=[[0.,0.],[2./3.,1./3.],[.5,.5],[0.,0.]]
# labels of the nodes
label=(r'$\Gamma$',r'$K$', r'$M$', r'$\Gamma$')
# total number of interpolated k-points along the path
nk=121

# call function k_path to construct the actual path
(k_vec,k_dist,k_node)=my_model.k_path(path,nk)
# inputs:
#   path, nk: see above
#   my_model: the pythtb model
# outputs:
#   k_vec: list of interpolated k-points
#   k_dist: horizontal axis position of each k-point in the list
#   k_node: horizontal axis position of each original node

print('-----')
print('starting calculation')
print('-----')
print('Calculating bands...')

# obtain eigenvalues to be plotted
evals=my_model.solve_all(k_vec)

# figure for bandstructure

fig, ax = plt.subplots()
# specify horizontal axis details
# set range of horizontal axis
ax.set_xlim(k_node[0],k_node[-1])
# put tickmarks and labels at node positions
ax.set_xticks(k_node)
ax.set_xticklabels(label)
# add vertical lines at node positions
for n in range(len(k_node)):
    ax.axvline(x=k_node[n],linewidth=0.5, color='k')
# put title
ax.set_title("Silicene band structure")
ax.set_xlabel("Path in k-space")
ax.set_ylabel("Band energy (eV)")

# plot first and second band
ax.plot(k_dist,evals[0])
ax.plot(k_dist,evals[1])

# make a PDF figure of a plot
fig.tight_layout()
fig.savefig("Silicene.pdf")

print('Done.\n')
```

Listing 1: Python code for electronic band structure computation of Silicene (same for Graphene except  $t = -3.033$ ).

## B APPENDIX 2

```
from __future__ import print_function
from pythtb import * # import TB model class
import numpy as np
import matplotlib.pyplot as plt

# define lattice vectors
lat=[[1.0,0.0],[0.5,np.sqrt(3.0)/2.0]]
# define coordinates of orbitals
orb=[[1./3.,1./3.],[2./3.,2./3.]]

# make two dimensional tight-binding graphene model
my_model=tb_model(2,2,lat,orb)

# set model parameters
delta=-0.1 # small staggered onsite term
t=-1.12

# set on-site energies
my_model.set_onsite([-delta,delta])
# set hoppings (one for each connected pair of orbitals)
# (amplitude, i, j, [lattice vector to cell containing j])
my_model.set_hop(t, 0, 1, [ 0, 0])
my_model.set_hop(t, 1, 0, [ 1, 0])
my_model.set_hop(t, 1, 0, [ 0, 1])

# print tight-binding model
my_model.display()

# construct circular path around Dirac cone
# parameters of the path
circ_step=31
circ_center=np.array([1.0/3.0,2.0/3.0])
circ_radius=0.05
# one-dimensional wf_array to store wavefunctions on the path
w_circ=wf_array(my_model,[circ_step])
# now populate array with wavefunctions
for i in range(circ_step):
    # construct k-point coordinate on the path
    ang=2.0*np.pi*float(i)/float(circ_step-1)
    kpt=np.array([np.cos(ang)*circ_radius,np.sin(ang)*circ_radius])
    kpt+=circ_center
    # find eigenvectors at this k-point
    (eval,evec)=my_model.solve_one(kpt,eig_vectors=True)
    # store eigenvector into wf_array object
    w_circ[i]=evec
# make sure that first and last points are the same
w_circ[-1]=w_circ[0]

# compute Berry phase along circular path
print("Berry phase along circle with radius: ",circ_radius)
print(" centered at k-point: ",circ_center)
print(" for band 0 equals      : ", w_circ.berry_phase([0],0))
print(" for band 1 equals      : ", w_circ.berry_phase([1],0))
print(" for both bands equals: ", w_circ.berry_phase([0,1],0))
print()

# construct two-dimensional square patch covering the Dirac cone
# parameters of the patch
square_step=31
square_center=np.array([1.0/3.0,2.0/3.0])
square_length=0.1
# two-dimensional wf_array to store wavefunctions on the path
w_square=wf_array(my_model,[square_step,square_step])
```

```
all_kpt=np.zeros((square_step,square_step,2))
# now populate array with wavefunctions
for i in range(square_step):
    for j in range(square_step):
        # construct k-point on the square patch
        kpt=np.array([square_length*(-0.5+float(i)/float(square_step-1)),
                      square_length*(-0.5+float(j)/float(square_step-1))])
        kpt+=square_center
        # store k-points for plotting
        all_kpt[i,j,:]=kpt
        # find eigenvectors at this k-point
        (eval,evec)=my_model.solve_one(kpt,eig_vectors=True)
        # store eigenvector into wf_array object
        w_square[i,j]=evec

# compute Berry flux on this square patch
print("Berry flux on square patch with length: ",square_length)
print("  centered at k-point: ",square_center)
print("  for band 0 equals      : ", w_square.berry_flux([0]))
print("  for band 1 equals      : ", w_square.berry_flux([1]))
print("  for both bands equals: ", w_square.berry_flux([0,1]))
print()

# also plot Berry phase on each small plaquette of the mesh
plaq=w_square.berry_flux([0],individual_phases=True)
#
fig, ax = plt.subplots()
ax.imshow(plaq.T,origin="lower",
          extent=(all_kpt[0,0,0],all_kpt[-2, 0,0],
                  all_kpt[0,0,1],all_kpt[ 0,-2,1]),)
ax.set_title("Berry curvature near Dirac cone of Silicene")
ax.set_xlabel(r"$k_x$")
ax.set_ylabel(r"$k_y$")
fig.tight_layout()
fig.savefig("Silicene_cone_phases.pdf")

print('Done.\n')
```

Listing 2: Python code for Berry flux computation of Silicene (same for Graphene except  $t = -3.033$ ).