

Experimental Physics (II)

Report

Experiment IV

Crystal And Reciprocal Space

Group 2

洪 瑜 B125090009

黃巧涵 B122030003

洪懌平 B102030019

2025/03/18

ABSTRACT

This week's experiment explores the fundamental concepts of crystal structures and reciprocal space through theoretical discussion and computational simulation. Using the example of graphene, i.e., a two-dimensional honeycomb lattice, we analyzed the roles of Bravais lattices, basis atoms, and lattice vectors in real space. Furthermore, we simulated Moiré patterns generated by twisted bilayer graphene using Python and analyzed how their periodicity varies with rotation angle. We also learned reciprocal space, deriving reciprocal lattice vectors, and visualizing diffraction patterns. These exercises deepened our understanding of crystal periodicity and its implications in diffraction phenomena.

1 Introduction

Crystals are materials whose internal structures exhibit a highly ordered and repeating pattern of atoms, ions, or molecules. This regularity can be described using a Bravais lattice, a set of discrete points generated through translational symmetry, and a basis, which defines the arrangement of atoms associated with each lattice point. Together, they form the full crystal structure.

In this experiment, we study the crystal structure of graphene, a monolayer of carbon atoms arranged in a two-dimensional hexagonal lattice. The lattice can be mathematically expressed using primitive lattice vectors, and its geometry gives rise to characteristic physical properties such as electronic behavior and diffraction patterns. A key concept in understanding such periodic structures is the reciprocal lattice, which resides in Fourier (momentum) space and is fundamental in interpreting X-ray diffraction, electron microscopy, and band structure analysis.

Through both analytical derivation and Python simulation, we investigate the real-space and reciprocal-space lattice vectors of graphene. In the advanced portion of the experiment, we explore Moiré patterns produced by twisting one layer of graphene relative to another, observing how small twist angles lead to large-scale interference patterns. These patterns highlight the intricate interplay between real and reciprocal space and are of contemporary interest in condensed matter physics, particularly in the study of superconductivity and topological materials.

In the following subsections, we will introduce the principles utilized in these exercises.

1.1 Crystal Structure, and Lattice, and Translational Symmetry

A crystal is a solid material whose atoms, molecules, or ions are arranged in a highly ordered repeating pattern extending in all spatial dimensions. This orderly structure is described using the concepts of a lattice and a basis, which together define the geometry of the crystal..

- **Lattice (point):**

A lattice is a regular array of points in space. These points represent repeating units and are defined mathematically using vectors.

- **basis:** The basis is the set of atoms associated with each lattice point. When the basis is added to every point in a lattice, it forms the complete crystal structure.

- **Bravais lattice:**

A special type of lattice in which every lattice point is equivalent. The environment surrounding any point looks identical when viewed from any other point, due to the crystal's translational symmetry.

- **Translation Vector:**

A crystal can be translated by a vector of the form

$$\vec{T} = n\vec{a}_1 + m\vec{a}_2 + l\vec{a}_3 \quad (1)$$

where n, m, l are integers and $\vec{a}_1, \vec{a}_2, \vec{a}_3$ are the primitive lattice vectors.

The repeating nature of crystals leads to **translation symmetry**, a property that ensures the crystal looks the same after being shifted by a lattice vector. This symmetry introduces many physical behaviors of crystalline materials and plays a key role in their diffraction patterns and electrical properties.

In two-dimensional crystals, only certain polygonal shapes can serve as unit cells that tessellate the plane without leaving gaps or overlaps. These are:

- Triangle
- Square
- Rectangle
- Parallelogram
- Rhombus (a special case of parallelogram)
- Hexagon

Other shapes, like regular pentagons and octagons, generally cannot tile space purely by translation since their internal angles are not a divisor of 360° (108° and 135° , respectively) unless specially deformed or combined with other shapes (e.g., [Pentagonal tiling](#)).

1.2 Graphene

Graphene is an ideal two-dimensional material that naturally illustrates the concepts of lattice, basis, and translation symmetry. It consists of a single layer of carbon atoms arranged in a hexagonal, or so-called honeycomb, structure.

The lattice properties of graphene:

- **Lattice Type:**

While the apparent honeycomb pattern of graphene, the underlying Bravais lattice of graphene is triangular. The honeycomb structure stems from the basis attached to each lattice point.

- **Unit Cell Shape:**

The repeating unit forms a parallelogram with two atoms per unit cell, consistent with the constraints of translation symmetry in 2D crystals.

(Other properties are discussed in the [Sec.3.1](#) and [4.1](#).)

1.3 Fourier transform

The **Fourier Transform (FT)** is a mathematical tool that converts a function or signal from the time domain to the frequency domain. In the time domain, a signal $f(t)$ varies with time. The FT decomposes the signal into a combination of sinusoidal components at different frequencies, allowing us to analyze its spectral characteristics.

For a continuous function, the Fourier Transform is defined as:

$$F(\omega) = \int_{-\infty}^{\infty} f(t)e^{-i\omega t} dt \quad (2)$$

where ω means frequency domain and $F(\omega)$ is the spectrum or FT of the $f(t)$.

On the contrary, by the Fourier inversion theorem, the spectrum $F(\omega)$ can be inversely transformed to the time domain, i.e.,

$$f(t) = \int_{-\infty}^{\infty} F(\omega)e^{i\omega t} dt \quad (3)$$

For example, consider a square pulse signal defined as:

$$f(t) = \begin{cases} A, & |t| \leq T/2 \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

and its FT is:

$$F(\omega) = T \operatorname{sinc}\left(\frac{\omega T}{2}\right) \quad (5)$$

where the sinc function is defined as:

$$\operatorname{sinc}(x) = \frac{\sin(x)}{x} \quad (6)$$

This means that the frequency spectrum of a square pulse is a sinc function, which has a main lobe and multiple diminishing side lobes. This phenomenon is similar to the result of single-slit diffraction because the slit aperture can be regarded as a "square" function in the spatial domain, and its diffraction pattern corresponds to its Fourier Transform.

Practically speaking, in the signal processing or engineering areas, **Discrete Fourier Transform (DFT)** or specifically **Fast Fourier Transform (FFT)** which includes efficient algorithms to calculate DFT is more widely used to perform FT to discrete and digital signal.

The DFT is defined as:

$$X_k = \sum_{n=0}^{N-1} x_n e^{-i2\pi kn/N} \quad (7)$$

for $k = 0, 1, 2, \dots, N-1$ where x_n is the discrete input signal, X_k is the output spectrum, N is the number of point in the signal.

In the same manner, DFT also has its inverse transformation to transform the discrete spectrum back to the discrete signal, i.e.,

$$x_n = \frac{1}{N} \sum_{k=0}^{N-1} X_k e^{i2\pi kn/N} \quad (8)$$

In Fig.1, The FFT result (blue curve) shows a sinc function, which matches the theoretical result (red dashed line). The width of the main lobe is inversely proportional to T (the pulse width), demonstrating the duality between time and frequency: the wider the time domain signal, the narrower the frequency domain response.

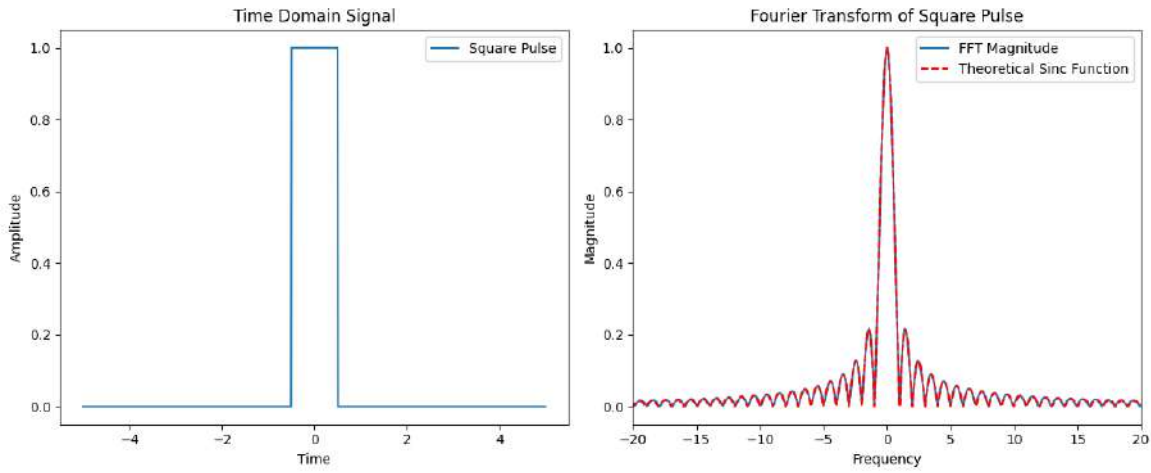


Figure 1: FFT of a square pulse using scipy.

1.4 Reciprocal Space

Reciprocal space is a concept in physics and materials science, commonly found in fields such as solid-state physics, X-ray diffraction, electron diffraction, and crystallography. It forms a dual relationship with real space and is used to describe wave vectors (such as light waves, electron waves, or sound waves) within a crystal. Definition: Reciprocal space is a mathematical space obtained from real space (the physical space of a crystal lattice) through a Fourier transform. In reciprocal space, the periodicity of a crystal is represented by the wavenumber (\mathbf{k}) or reciprocal lattice vector (\mathbf{G}), which are inversely related to the lattice constants in real space.

If the lattice in real space has basis vectors \mathbf{a}_1 , \mathbf{a}_2 , \mathbf{a}_3 , then the corresponding reciprocal lattice basis vectors \mathbf{b}_1 , \mathbf{b}_2 , \mathbf{b}_3 are determined by the following relationships:

$$\mathbf{b}_1 = 2\pi \frac{\mathbf{a}_2 \times \mathbf{a}_3}{\mathbf{a}_1 \cdot (\mathbf{a}_2 \times \mathbf{a}_3)} \quad (9)$$

$$\mathbf{b}_2 = 2\pi \frac{\mathbf{a}_3 \times \mathbf{a}_1}{\mathbf{a}_2 \cdot (\mathbf{a}_3 \times \mathbf{a}_1)} \quad (10)$$

$$\mathbf{b}_3 = 2\pi \frac{\mathbf{a}_1 \times \mathbf{a}_2}{\mathbf{a}_3 \cdot (\mathbf{a}_1 \times \mathbf{a}_2)} \quad (11)$$

These reciprocal lattice basis vectors satisfy the following relationship:

$$\mathbf{a}_i \cdot \mathbf{b}_j = 2\pi \delta_{ij} \quad (12)$$

Physical significance:

- **X-ray Diffraction:**

Reciprocal space describes the Fourier components of a crystal structure. The Bragg diffraction condition is often expressed in reciprocal space as:

$$\mathbf{k}' - \mathbf{k} = \mathbf{G} \quad (13)$$

where \mathbf{k} and \mathbf{k}' are the wave vectors of the incident and scattered light, respectively, and \mathbf{G} is the reciprocal lattice vector.

- **Electronic Band Structure:** In solid-state physics, the wave nature of electrons is closely related to reciprocal space through Bloch's theorem. Band structure diagrams are often plotted in reciprocal space.
- **Fourier Transform:**
The electron density and atomic arrangement of materials can be transformed into reciprocal space using Fourier transforms, facilitating spectral analysis.

1.5 Diffraction and interference

- **Diffraction:**
The bending and spreading of waves when they encounter an obstacle or pass through a narrow opening.
- **Interference:**
The overlapping of two or more waves leads to constructive or destructive patterns.
- **Single-Slit Diffraction Pattern:**
The intensity distribution follows a sinc^2 function:

$$I(\theta) = I_0 \left(\frac{\sin(\pi a \sin \theta / \lambda)}{\pi a \sin \theta / \lambda} \right)^2 = I_0 \text{sinc}^2(\pi a \sin \theta / \lambda) \quad (14)$$

This pattern closely resembles the Fourier transform of a square pulse, which is also a sinc function. This is because the single-slit aperture acts as a spatial filter, and its diffraction pattern is its Fourier transform.

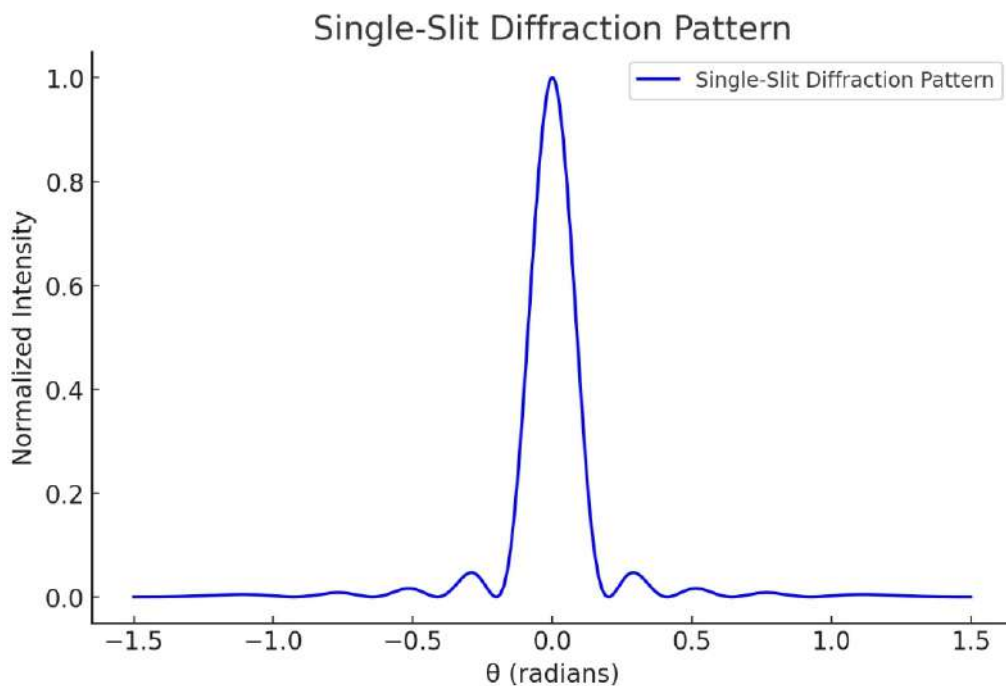


Figure 2: Single-slit diffraction pattern

2 Experimental steps

2.1 Graphene Crystal Structure

1. Describe the type of lattice present in graphene.
2. Identify the basis atoms in the unit cell of graphene.
3. How many carbon atoms constitute the basis?
4. Express the lattice vectors.
5. Use these vectors to generate graphene lattice via Python.

Bonus:

1. Use MATLAB/Python to create a twisted bilayer graphene Moiré pattern (in real space).
2. Find the relationship between the periodicity of the Moiré pattern and its twist angles.

2.2 Reciprocal Space

1. Please draw the reciprocal lattice vector G in Fig.3(b) and define the length of G .

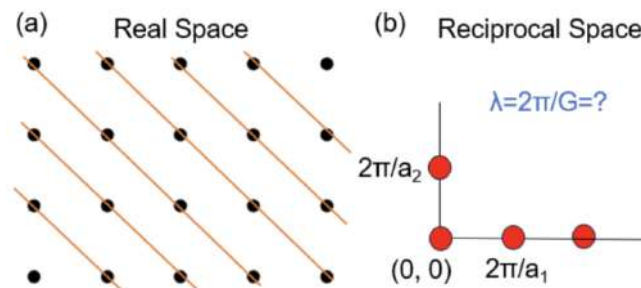


Figure 3: Demonstration of reciprocal space (credit: Experiment Guideline)

2. Use the relationship $(\mathbf{b}_i \cdot \mathbf{a}_j = 2\pi\delta_{ij})$ where $\delta_{ij} = 1$ if $i = j$ and $\delta_{ij} = 0$ if $i \neq j$ to obtain the reciprocal lattice vectors of graphene and again use python to generate the reciprocal lattice point of graphene. Find the diffraction patterns of graphene from a website and define the reciprocal lattice vectors. Compare these obtained vectors with your results.
3. Use the lattice points and the reciprocal lattice vectors you obtained from Python to check if reciprocal lattice points can be mapped out by the set of reciprocal lattice vectors that yield plane waves with the periodicity of a given Bravais lattice.

3 Result

3.1 Graphene Crystal Structure

1. **Describe the type of lattice present in graphene.**

Graphene has a honeycomb lattice structure, which can be described as a hexagonal (2D) Bravais lattice with a two-atom basis. The underlying lattice is a triangular Bravais lattice, but the honeycomb structure emerges due to the presence of two inequivalent carbon atoms (A and B) in each unit cell.

2. **Identify the basis atoms in the unit cell of graphene.**

Graphene's unit cell consists of two carbon atoms, labeled as A and B.

- **Atom A:** Located at $(0, 0)$
- **Atom B:** Located at $\left(\frac{a}{2}, \frac{\sqrt{3}a}{6}\right)$, where a is the nearest-neighbor bond length

3. **How many carbon atoms constitute the basis?**

The basis of graphene consists of two carbon atoms per unit cell.

4. **Express the lattice vectors.**

Graphene's **primitive lattice vectors** (\mathbf{a}_1 and \mathbf{a}_2) in Cartesian coordinates are:

$$\mathbf{a}_1 = \left(\frac{3a}{2}, \frac{\sqrt{3}a}{2} \right) \quad (15)$$

$$\mathbf{a}_2 = \left(\frac{3a}{2}, -\frac{\sqrt{3}a}{2} \right) \quad (16)$$

where a is the **nearest-neighbor bond length** (approximately 1.42\AA), and the **lattice constant** a_0 (distance between two adjacent unit cells) is given by:

$$a_0 = \sqrt{3}a \approx 2.46\text{\AA} \quad (17)$$

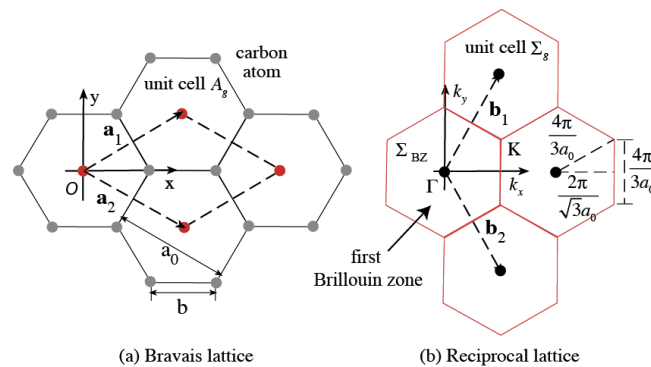


Figure 4: The structure of graphene. (a) Bravais lattice; (b) reciprocal lattice. [1]

5. Use these vectors to generate graphene lattice via Python.

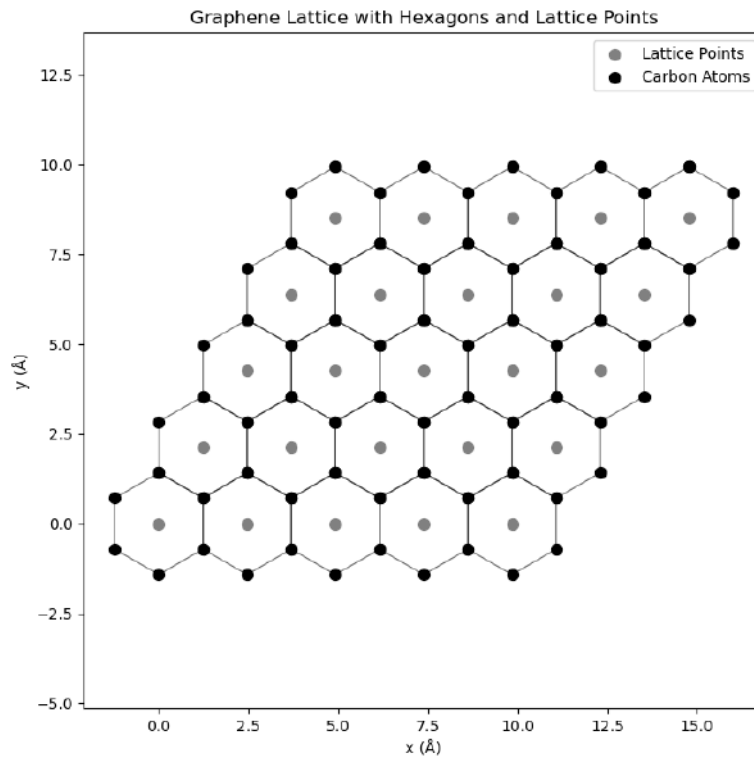


Figure 5: Lattice points of graphene

Bonus:

1. Use MATLAB/Python to create a twisted bilayer graphene Moiré pattern (in real space).

- 1°

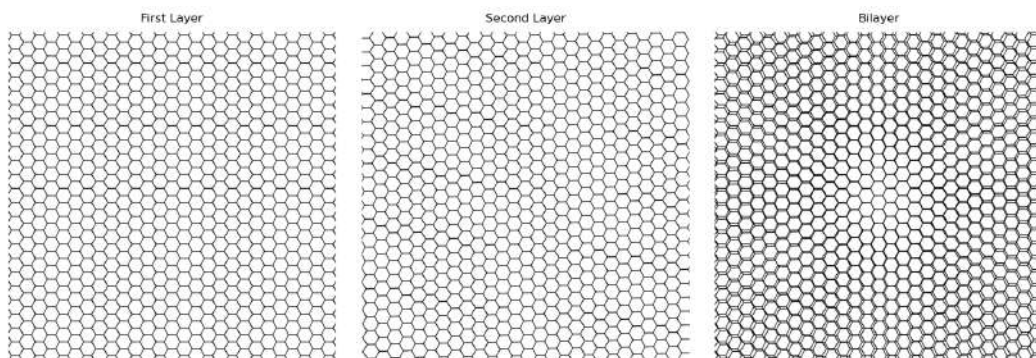


Figure 6: A twisted bilayer graphene Moiré pattern (1°)

2. 5°

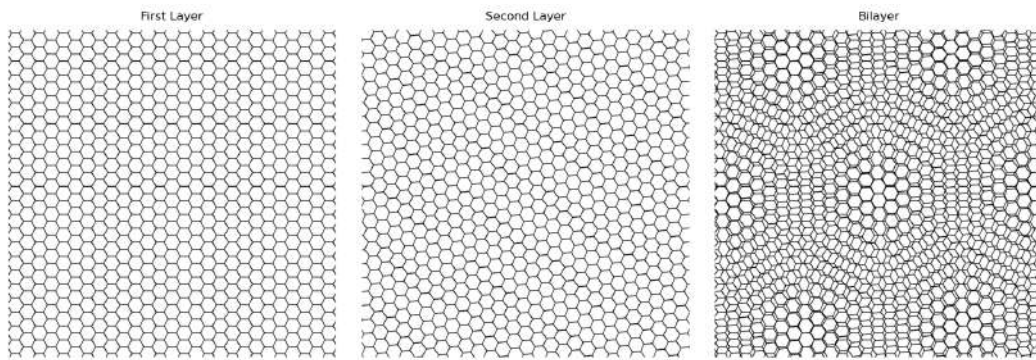


Figure 7: *continued* (5°)

3. 10°

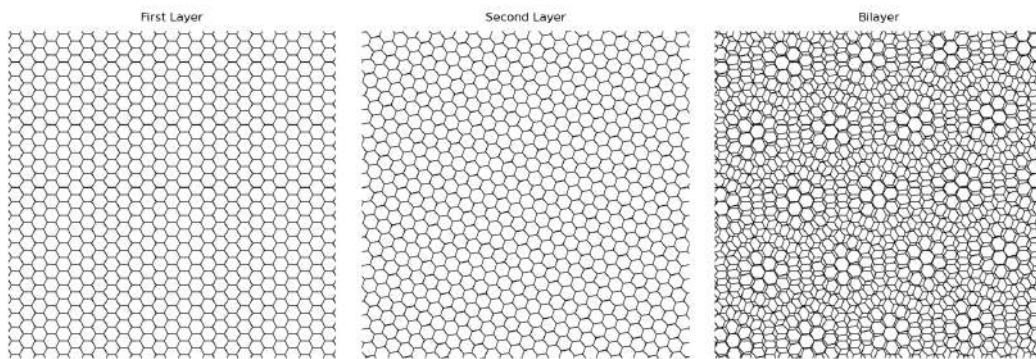


Figure 8: *continued* (10°)

• 15°

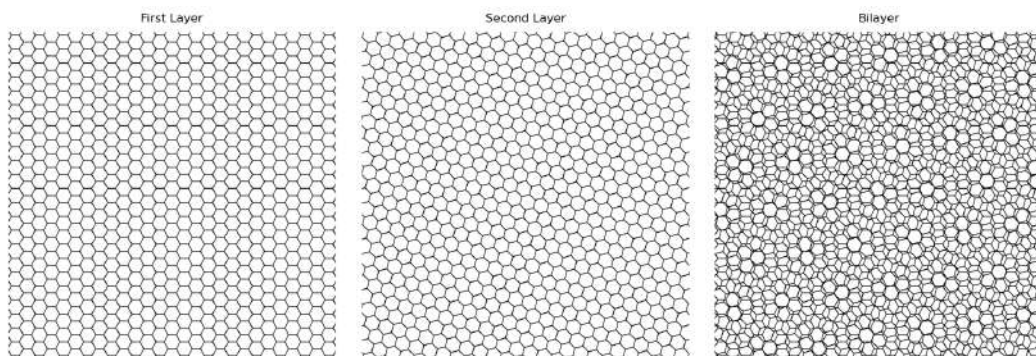
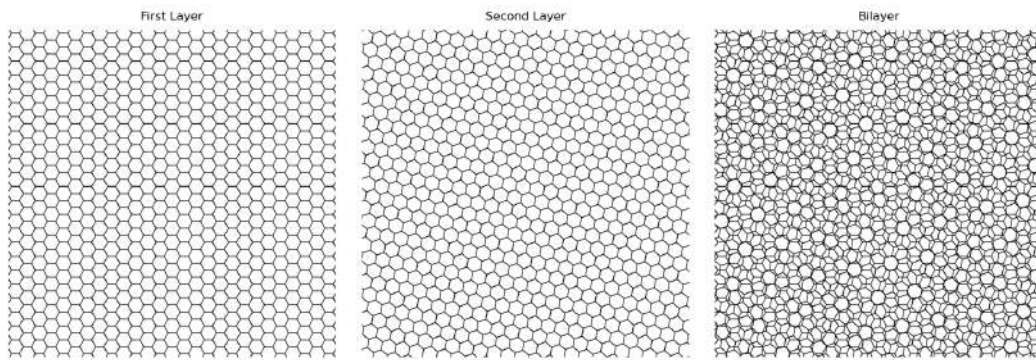
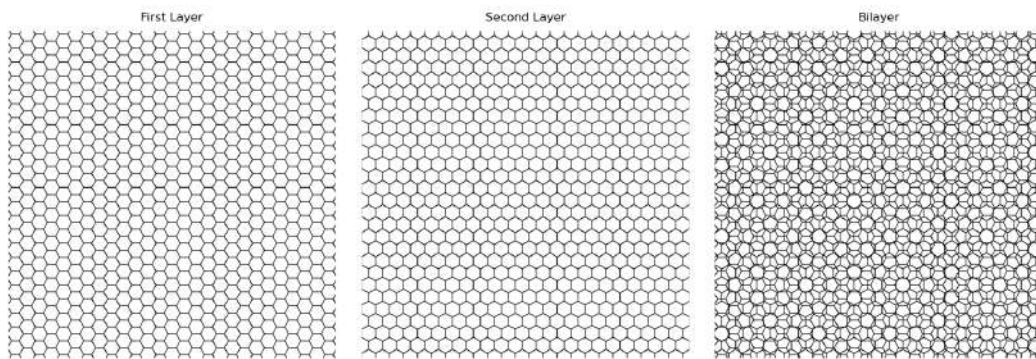
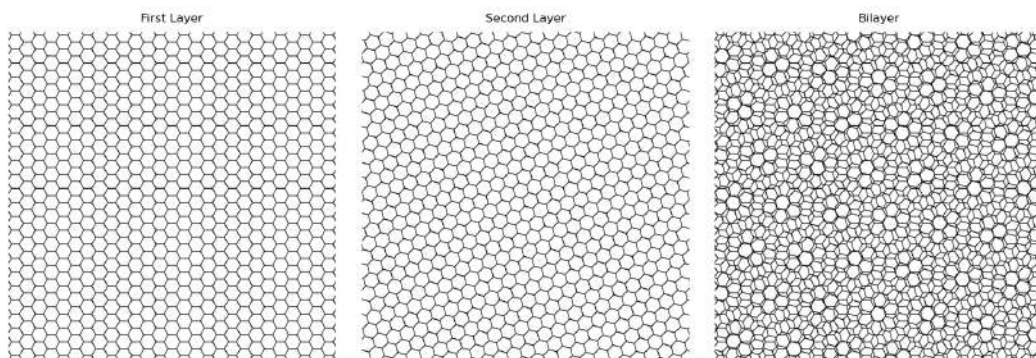
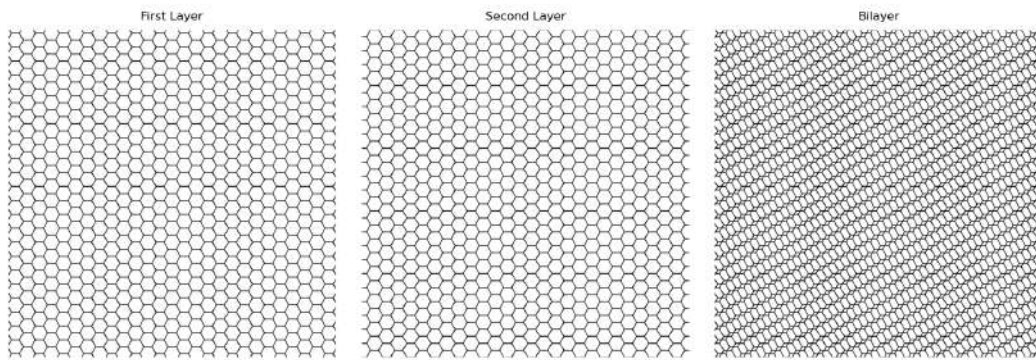
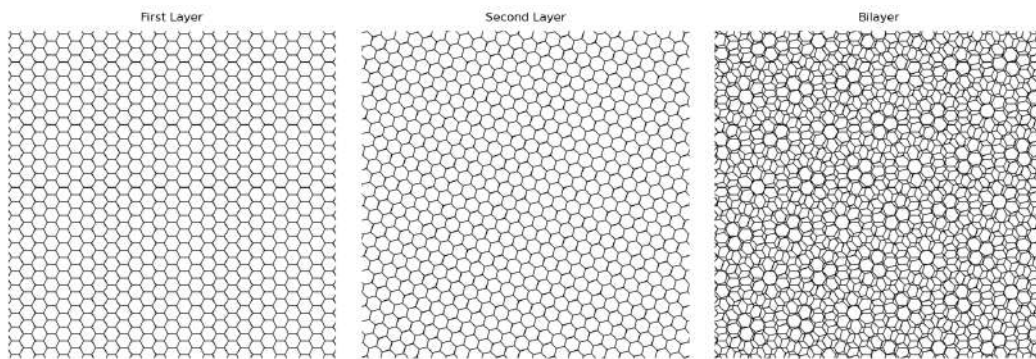
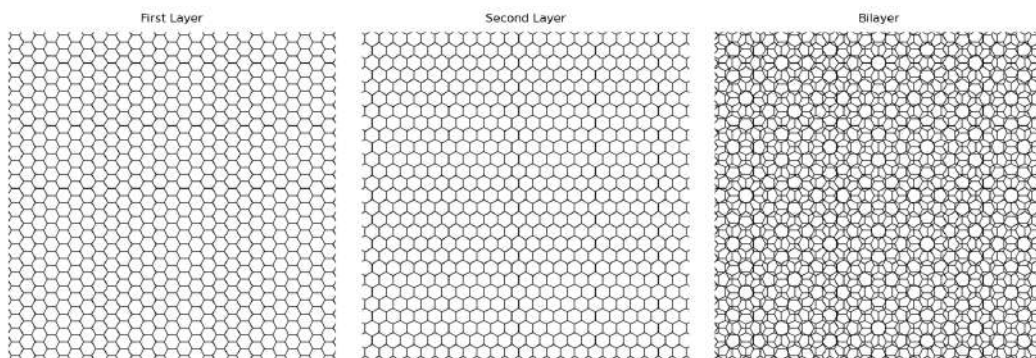


Figure 9: *continued* (15°)

4. 20° Figure 10: *continued* (20°)5. 30° Figure 11: *continued* (30°)• 45° Figure 12: *continued* (45°)

6. 60° Figure 13: *continued* (60°)7. 75° Figure 14: *continued* (75°)Figure 15: *continued* (90°)

8. Find the relationship between the periodicity of the Moiré pattern and its twist angles.

When the lattice constant of the two layers of graphene (a_0 , approximately 2.46 Å) and their twist angle (θ , assuming θ is very small) are given, the Moiré pattern periodicity L can be approximated as:

$$L \approx \frac{a_0}{2\sin(\theta/2)} \quad (18)$$

where L = Moiré pattern periodicity, a_0 is graphene lattice constant (approximately 2.46 Å), and θ is twist angle (in radians)

As the twist angle θ decreases, the Moiré pattern periodicity L increases.

(A detailed analysis is in the Sec. 4.1.)

3.2 Reciprocal Space

1. Please draw the reciprocal lattice vector \mathbf{G} in Fig.3(b) and define the length of \mathbf{G} .

(There are a few corrections from the experiment notebook, e.g., Fig. 7 in the notebook.)

The **reciprocal lattice vector**:

$$\mathbf{G} = \frac{2\pi}{a_1}\hat{\mathbf{i}} + \frac{2\pi}{a_2}\hat{\mathbf{j}} \quad (19)$$

$$|\mathbf{G}| = 2\pi \sqrt{\left(\frac{1}{a_1}\right)^2 + \left(\frac{1}{a_2}\right)^2} \quad (20)$$

The **wavelength**:

$$\lambda = \frac{2\pi}{|\mathbf{G}|} = \frac{1}{\sqrt{\left(\frac{1}{a_1}\right)^2 + \left(\frac{1}{a_2}\right)^2}} \quad (21)$$

2. Use the relationship ($\mathbf{b}_i \cdot \mathbf{a}_i = 2\pi\delta_{ij}$ where $\delta_{ij} = 1$ if $i = j$ and $\delta_{ij} = 0$ if $i \neq j$) to obtain the reciprocal lattice vectors of graphene and again use python to generate the reciprocal lattice point of graphene. Find the diffraction patterns of graphene from a website and define the reciprocal lattice vectors. Compare these obtained vectors with your results.

(a) Use Python to generate the reciprocal lattice of graphene and calculate its \mathbf{b}_1 and \mathbf{b}_2 :

- i. This part simulates the honeycomb lattice of graphene, where the angle between the two basis vectors is 120° . The basis vectors, \mathbf{a}_1 and \mathbf{a}_2 form the lattice vectors in real space.

$$\mathbf{a}_1 = (0, 0) \quad (22)$$

$$\mathbf{a}_2 = a \cdot (\cos(120^\circ), \sin(120^\circ)) = \left(-\frac{a}{2}, \frac{\sqrt{3}a}{2}\right) \quad (23)$$

```
import numpy as np
import matplotlib.pyplot as plt

# 1. Real space lattice vectors for graphene (hexagonal lattice)
a1 = np.array([1.0, 0.0]) # Arbitrary unit
angle = 120 * np.pi / 180 # 120 degrees in radians
a2 = np.array([np.cos(angle), np.sin(angle)])
```

Figure 16: Real Space Lattice Vectors

- ii. We use the cross product of vectors to determine the reciprocal lattice, which must be handled in three-dimensional space.

```
# 2. Calculate reciprocal lattice vectors using  $\mathbf{b}_i \cdot \mathbf{a}_j = 2\pi \delta_{ij}$ 
# Formula:  $\mathbf{b}_1 = 2\pi \cdot (\mathbf{a}_2 \times \mathbf{z}) / (\mathbf{a}_1 \times \mathbf{a}_2)$ 
z_hat = np.array([0, 0, 1])
a1_3d = np.array([a1[0], a1[1], 0])
a2_3d = np.array([a2[0], a2[1], 0])
```

Figure 17: Reciprocal Lattice Vectors

- iii. According to the definition, i.e., $\mathbf{b}_i \cdot \mathbf{a}_j = 2\pi\delta_{ij}$, the reciprocal lattice vectors are determined as following:

$$\mathbf{b}_1 = \frac{2\pi(\hat{\mathbf{z}} \times \mathbf{a}_2)}{|\mathbf{a}_1 \times \mathbf{a}_2|} = \left(\frac{2\pi}{a}, \frac{2\pi}{\sqrt{3}a} \right) \quad (24)$$

$$\mathbf{b}_2 = \frac{2\pi(\mathbf{a}_1 \times \hat{\mathbf{z}})}{|\mathbf{a}_1 \times \mathbf{a}_2|} = \left(0, \frac{4\pi}{\sqrt{3}a} \right) \quad (25)$$

```
# Cross product and area
area = np.linalg.norm(np.cross(a1_3d, a2_3d))
b1_3d = 2 * np.pi * np.cross(z_hat, a2_3d) / area
b2_3d = 2 * np.pi * np.cross(a1_3d, z_hat) / area
```

Figure 18: Cross product and area

- iv. Project them back into 2D to obtain the reciprocal lattice vectors b_1 and b_2 for visualization.

```
# Project to 2D
b1 = b1_3d[:2]
b2 = b2_3d[:2]
```

Figure 19: Project to 2D

- v. Use a nested “for loop” to generate all reciprocal lattice points through the linear combination “ $n*b_1+m*b_2$ ”

```
# 3. Generate extended reciprocal lattice points with margin
n_range = range(-5, 6) # Extended further to ensure edge coverage
reciprocal_points = []
for n in n_range:
    for m in n_range:
        point = n * b1 + m * b2
        reciprocal_points.append(point)
reciprocal_points = np.array(reciprocal_points)
```

Figure 20: Reciprocal Lattice Points

- vi. Initialize the plot and draw all reciprocal lattice points, marking them with blue dots.

```
# 4. Plot extended reciprocal lattice with parallelogram unit cells
plt.figure(figsize=(10, 10))
plt.scatter(reciprocal_points[:, 0], reciprocal_points[:, 1], color='blue', s=10, label='Reciprocal Lattice Points')
```

Figure 21: Plot extended reciprocal lattice with parallelogram unit cells

- vii. Check whether neighboring lattice points exist to avoid drawing incomplete unit cells beyond the boundaries. Use orange dashed lines to outline the complete parallelogram unit cell.

```
# Filter points to avoid drawing partial parallelograms on edges
for point in reciprocal_points:
    neighbors = [point + b1, point + b1 + b2, point + b2]
    if all(any(np.allclose(p, rp) for rp in reciprocal_points) for p in neighbors):
        vertices = np.array([point, point + b1, point + b1 + b2, point + b2, point])
        plt.plot(vertices[:, 0], vertices[:, 1], color='orange', linestyle='--', linewidth=0.5)
```

Figure 22: Filter points to avoid drawing partial parallelograms on edges

- viii. First, use arrows to indicate the reciprocal lattice vectors \mathbf{b}_1 and \mathbf{b}_2 . Then, finalize the visualization by adding a title, coordinate axes, and grid lines, ensuring a consistent aspect ratio and displaying a legend.

```
plt.quiver(0, 0, b1[0], b1[1], angles='xy', scale_units='xy', scale=1, color='red', label='b1')
plt.quiver(0, 0, b2[0], b2[1], angles='xy', scale_units='xy', scale=1, color='green', label='b2')
plt.title('Extended Graphene Reciprocal Lattice with Parallelogram Grid')
plt.xlabel('kx')
plt.ylabel('ky')
plt.grid(True, linestyle=':')
plt.axis('equal')
plt.legend()
plt.show()
```

Figure 23: Refine the image for clarity and presentation

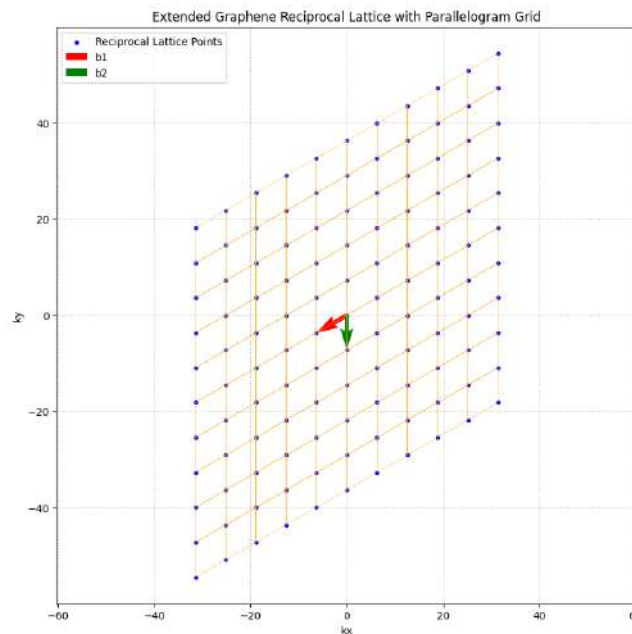


Figure 24: Reciprocal lattice of graphene

(b) Compare with the experimental diffraction pattern:

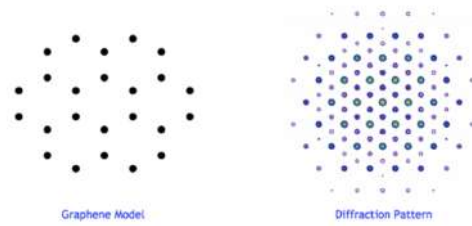


Figure 25: *left*: theoretical graphene model. Black dots represent carbon atom positions, forming a hexagonal arrangement characterized by 120° angles and sixfold symmetry; *right*: Diffraction pattern of the graphene model [2]



Figure 26: Graphene electron diffraction pattern [3]

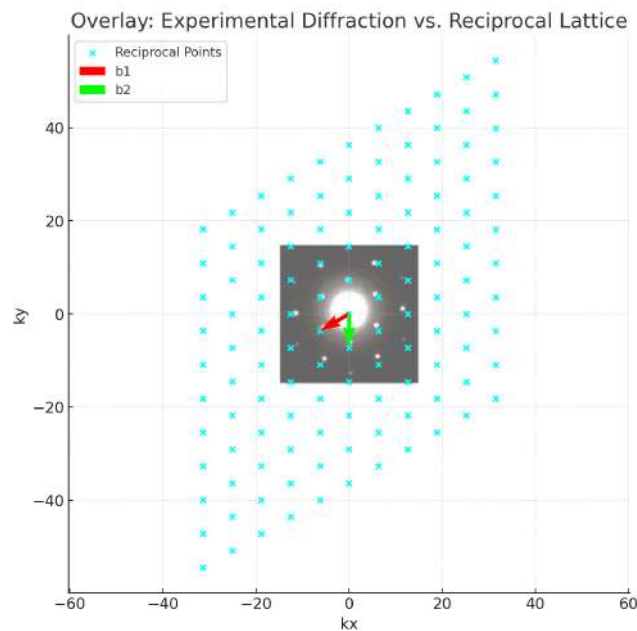


Figure 27: Compare the Python-generated plot with the experimental diffraction pattern

3. Use the lattice points and the reciprocal lattice vectors you obtained from Python to check if reciprocal lattice points can be mapped out by the set of reciprocal lattice vectors that yield plane waves with the periodicity of a given Bravais lattice.

The reciprocal lattice points can be generated through a linear combination:

$$\mathbf{G}_{mn} = m\mathbf{b}_1 + n\mathbf{b}_2 \quad (26)$$

where \mathbf{G} is the reciprocal lattice vector. If the reciprocal lattice points can form plane waves, they must satisfy the periodicity of the Bravais lattice, which is linked to the Brillouin zone and diffraction patterns.

- In a Bravais lattice, the reciprocal lattice vectors should yield plane waves that have the same periodicity as the real-space lattice.
- The plane wave is given by:

$$\psi(\mathbf{r}) = e^{i\mathbf{G} \cdot \mathbf{r}} \quad (27)$$

where the length of \mathbf{G} is $\frac{4\pi}{3a}$

- The reciprocal lattice vectors confirm that plane waves generated by them satisfy the periodicity of the real-space Bravais lattice.

(A detailed analysis is in the Sec. 4.2.)

4 Analysis and Discussion

4.1 Graphene Crystal Structure

Find the relationship between the periodicity of the Moiré pattern and its twist angles.

To estimate the periodicity of the Moiré pattern, we firstly use the FFT of the first, second, and the bilayer to (f_1 , f_1 , and $f_{bilayer}$) obtain their spectra, i.e., F_1 , F_1 , and $F_{bilayer}$, respectively. Then, extract the spectrum of the Moiré pattern by

$$F_{Moiré} = F_{bilayer} - F_1 - F_2 \quad (28)$$

- 1°

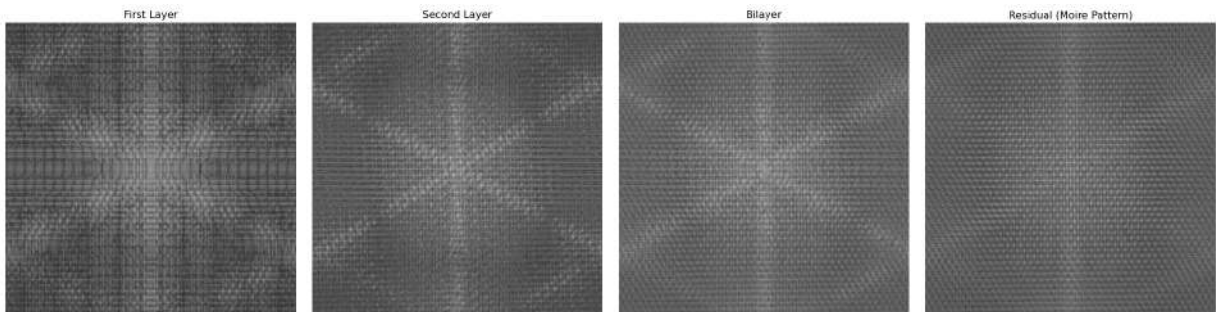
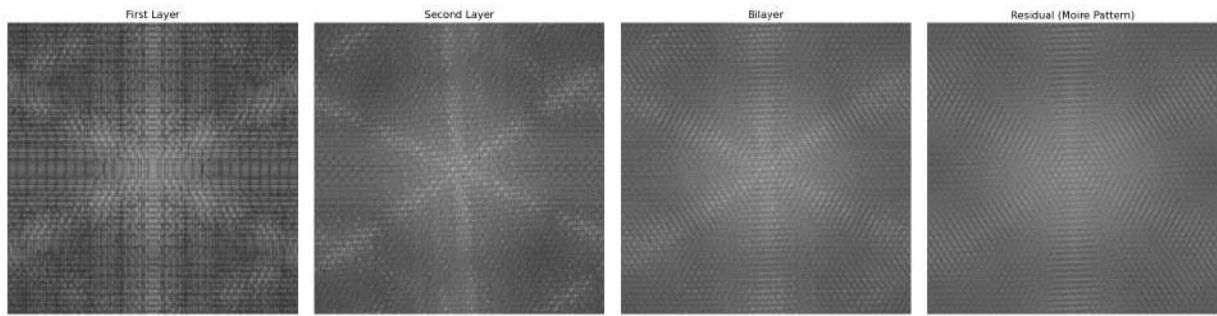
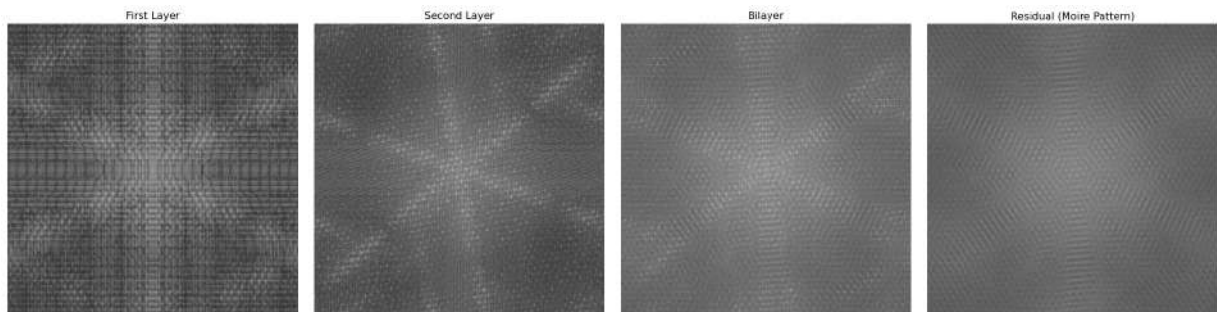
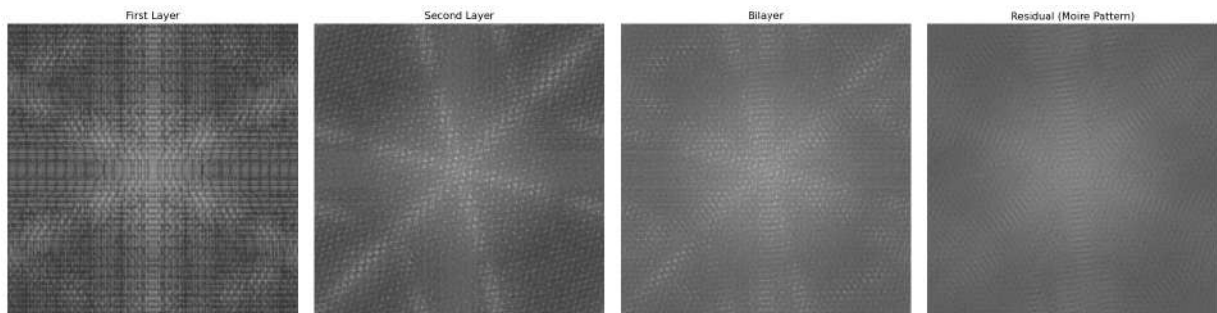


Figure 28: FFT of the first (*left*), second (*mid-left*), and the bilayer (*mid-right*). Also, we get the residual (*right*), the spectrum of the Moiré pattern, by subtracting the spectra of the bilayer from the ones of the first and second layer (1°)

• 5° Figure 29: *continued* (5°)• 10° Figure 30: *continued* (10°)• 15° Figure 31: *continued* (15°)

- 20°

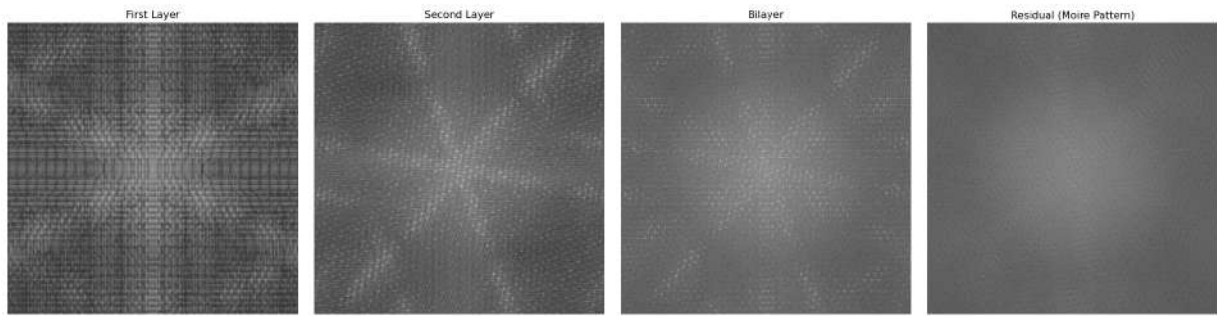


Figure 32: *continued* (20°)

- 30°

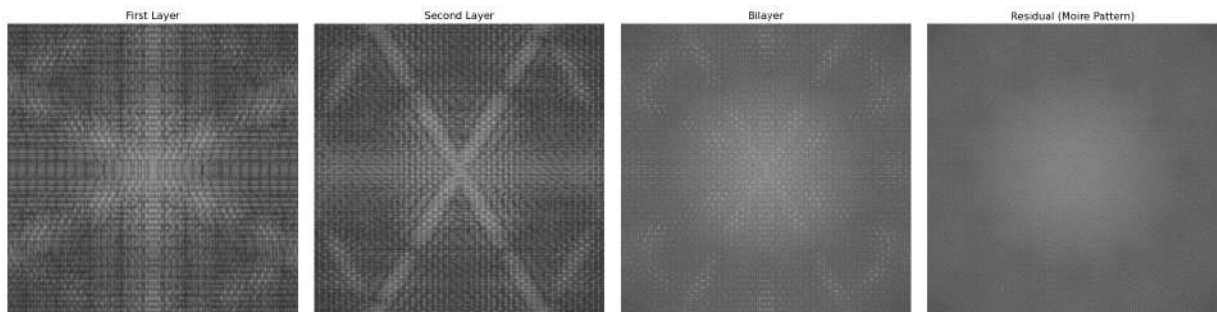


Figure 33: *continued* (30°)

- 45°

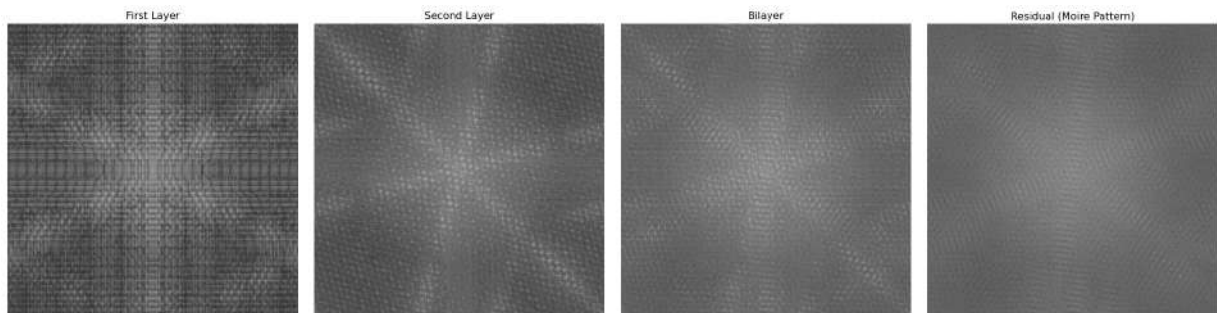
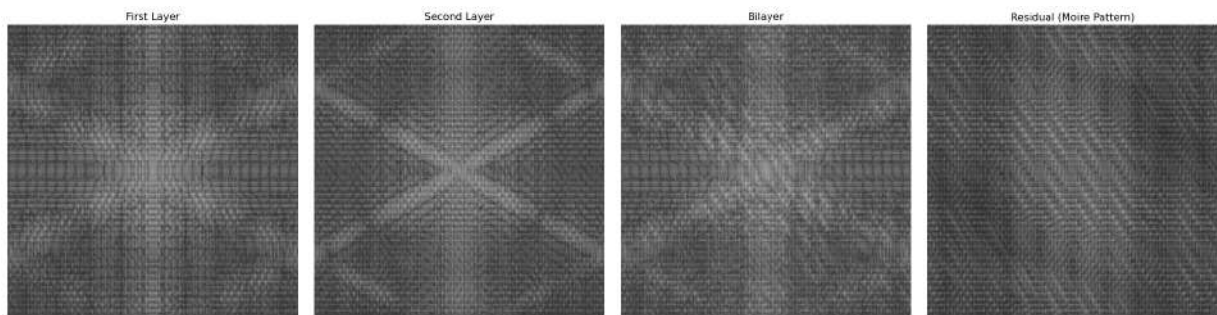
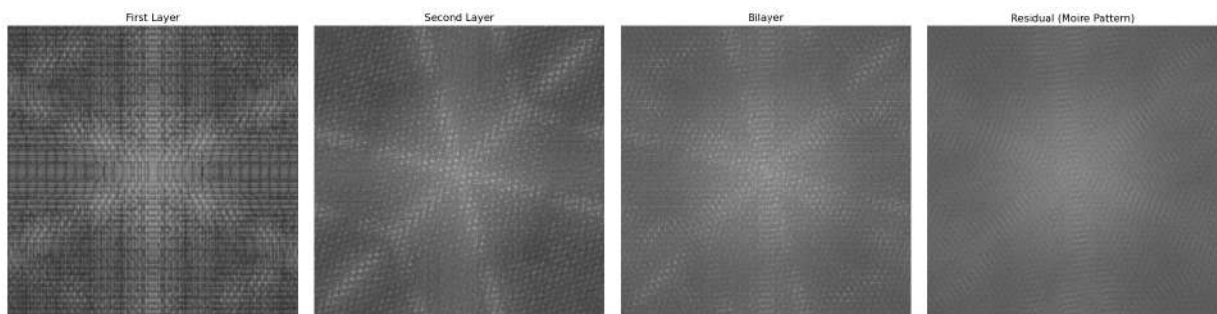


Figure 34: *continued* (45°)

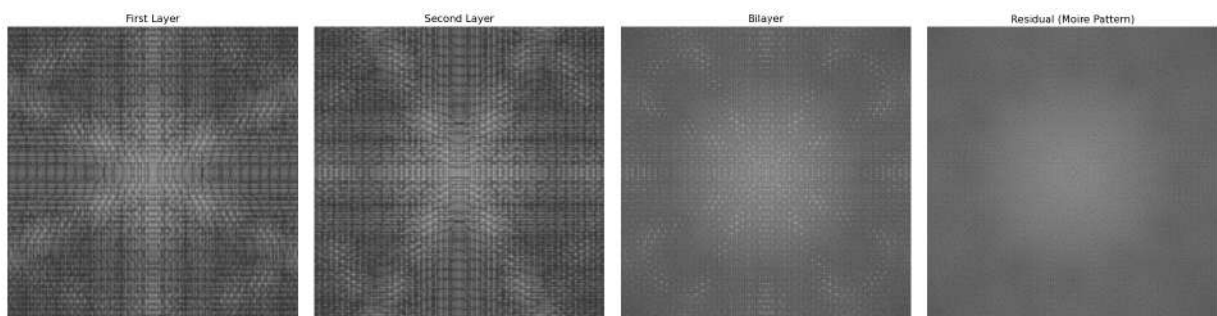
- 60°

Figure 35: *continued* (60°)

- 75°

Figure 36: *continued* (75°)

- 90°

Figure 37: *continued* (90°)

(Note: These spectra should be calculated using complex components to be mathematically correct.)

Secondly, inverse FFT the spectrum of the Moiré pattern $F_{Moiré}$ back to real space $f_{Moiré}$.

- 1°

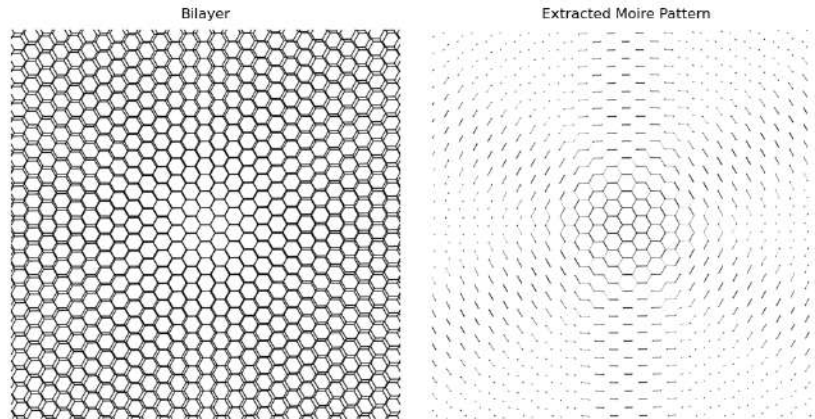


Figure 38: *left*: The Moiré pattern in the bilayer; *right*: Extracted Moiré pattern(1°)

- 5°

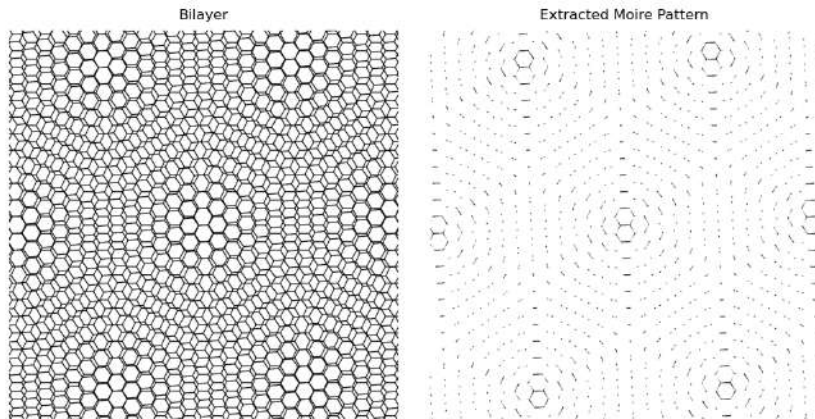


Figure 39: *continued*(5°)

- 10°

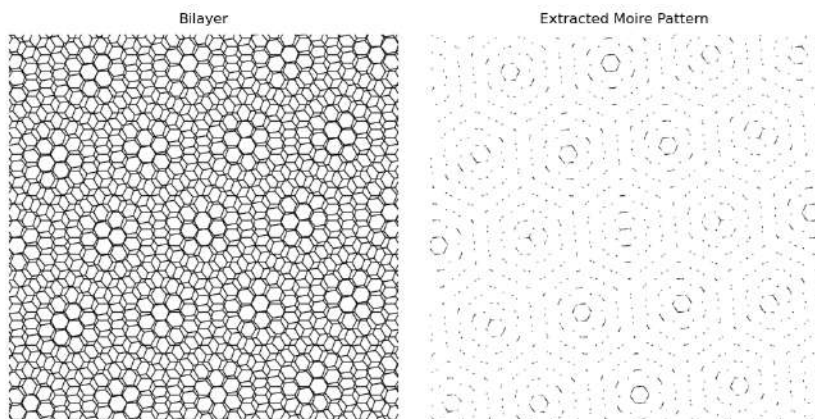


Figure 40: *continued*(10°)

- 15°

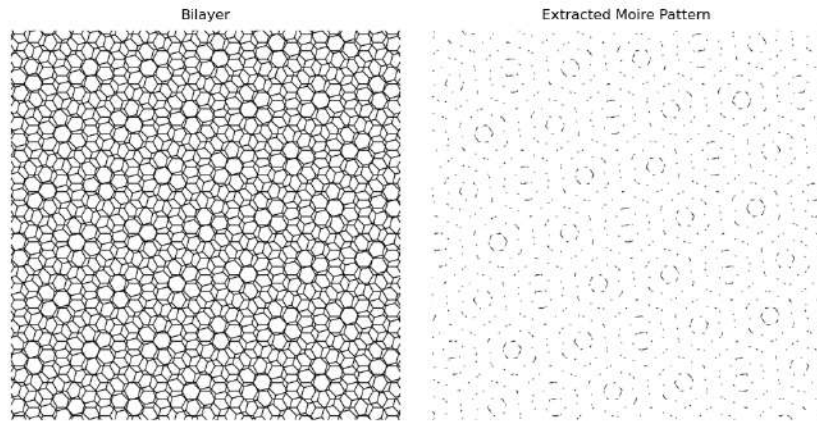


Figure 41: *continued*(15°)

- 20°

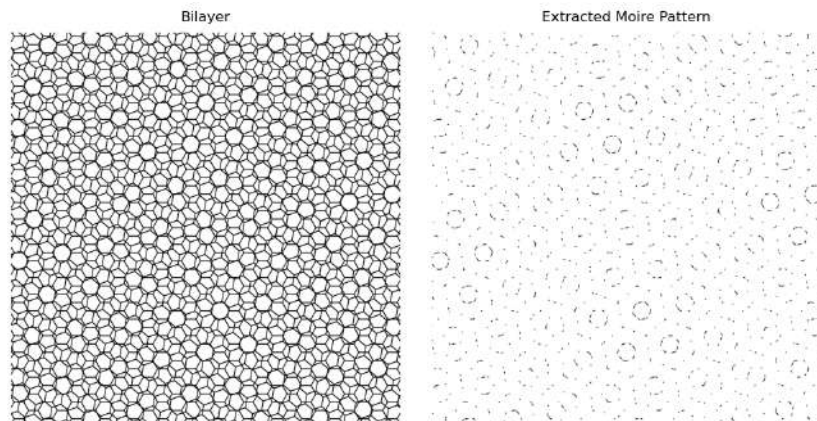


Figure 42: *continued*(20°)

- 30°

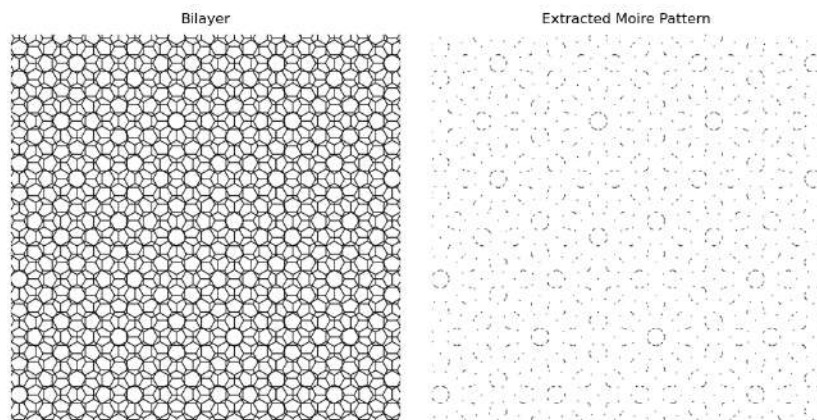


Figure 43: *continued*(30°)

- 45°

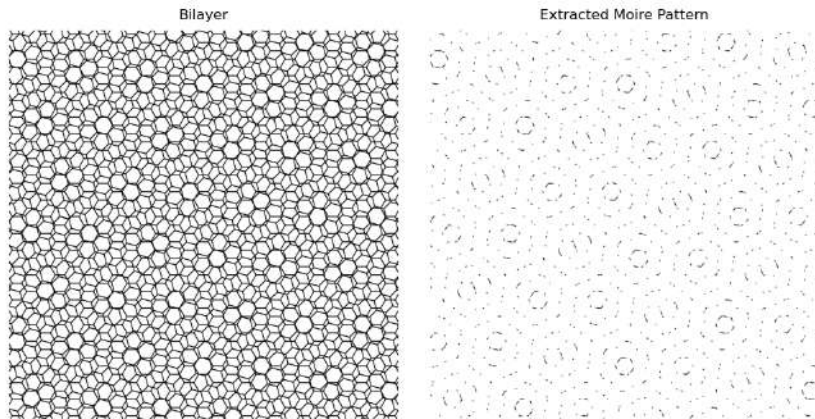


Figure 44: *continued*(45°)

- 60°

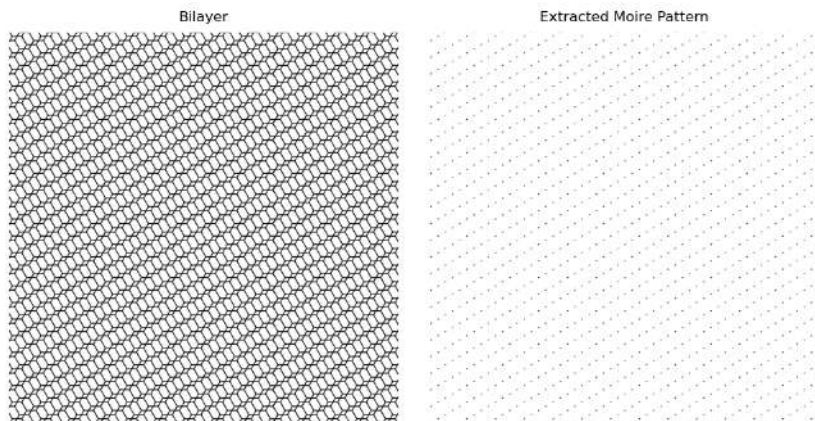


Figure 45: *continued*(60°)

- 75°

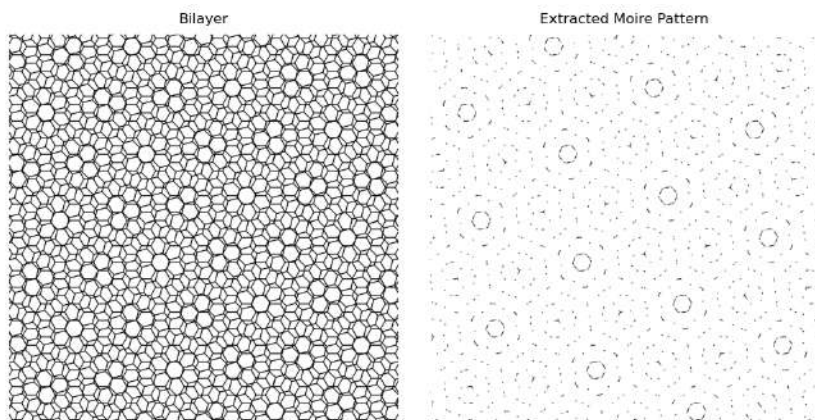
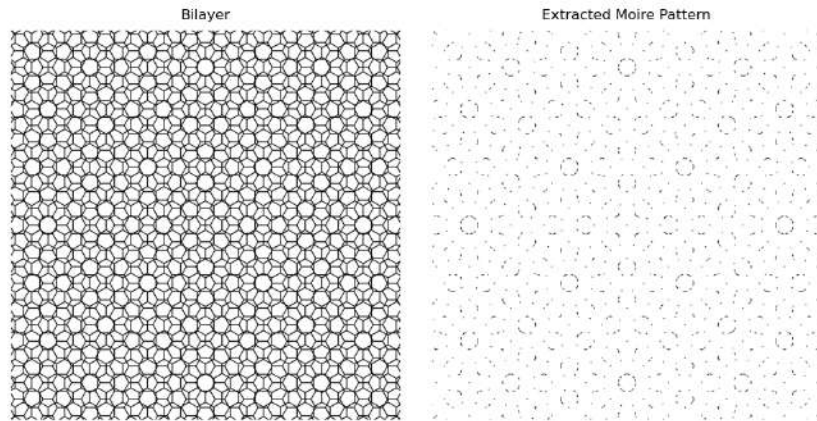


Figure 46: *continued*(75°)

- 90°

Figure 47: *continued*(90°)

Finally, estimate the periodicity of the extracted Moiré pattern by measuring the number of inter-period pixels and calculating the periodicity by Eq.18. ($a = 12\text{pixels} = 1.42\text{\AA}$, $a_0 = \sqrt{3}a$)

Twisted angle ($^\circ$)	Measured Periodicity (\AA)	Ideal Periodicity (\AA)	Measuring Error %
1	(too large to be measured)	57.297	-
5	10.593	11.463	7.59
10	5.612	5.737	2.18
15	3.682	3.831	3.89
20	2.573	2.879	8.54
30	-	1.932	-
45	3.584	1.307	174.21
60	-	1	-
75	4.081	0.821	397.07
90	-	0.707	-

As we can see in the table, if the twisted angle is $5^\circ - 20^\circ$, the measured periodicities are nearly the same as the predicted periodicity calculated by Eq.18. However, the measured periodicities seem "saturated" and become immeasurable if the twisted angle is $\theta \geq 30^\circ$, here are some possible reasons:

- Eq.18 is only accurate for small angles.
- The periodicities of the actual Moiré patterns are smaller than the bond lengths, inducing "**aliasing**" (recalling "Nyquist Theorem and Aliasing" last week) when doing FFT. Hence, the extracted Moiré patterns are not fully reconstructed.
- When $\theta = 60^\circ$, the twisted layer matches the 6-fold symmetry of the graphene, i.e., rotating by $\theta = 60^\circ$ brings the lattice back to itself. Therefore, as Fig.45 shown, no Moiré pattern is extracted.

4.2 Reciprocal Space

1. Please draw the reciprocal lattice vector \mathbf{G} in Fig.3(b) and define the length of \mathbf{G} .

The reciprocal space refers to the wavenumber space, where the unit represents "the number of wave peaks per unit length". $|\mathbf{G}|$ denotes the wavenumber (the number of phases per unit length), while the wavelength represents the spatial period of the wave. The relationship between the two can be referenced in the equation shown in Fig.3(b).

From the calculation results, we can observe the following:

- **The relationship between $|\mathbf{G}|$ and a_1, a_2 :**
As a_1 and a_2 increase, $|\mathbf{G}|$ decreases, indicating that when the lattice is sparse, the reciprocal lattice becomes denser. A smaller \mathbf{G} corresponds to a longer wavelength.
- **The limiting case of wavelength:**
if $a_1 = a_2 = a$, then $\lambda = a/\sqrt{2}$, meaning that the shortest wavelength in a specific direction is $a/\sqrt{2}$.

Knowing these results, we can infer the following physical properties:

- **Material structure analysis:**
Given λ , we can derive $|\mathbf{G}|$, then infer a_1, a_2 to determine whether the crystal structure is uniform.
 - **Directional dependence:**
Since λ varies when $a_1 \neq a_2$, it indicates that λ is direction-dependent. This allows us to analyze whether a crystal exhibits stronger or weaker properties (e.g., conductivity, optical properties) in certain directions.
2. Use the relationship ($\mathbf{b}_i \cdot \mathbf{a}_i = 2\pi\delta_{ij}$ where $\delta_{ij} = 1$ if $i = j$ and $\delta_{ij} = 0$ if $i \neq j$) to obtain the reciprocal lattice vectors of graphene and again use python to generate the reciprocal lattice point of graphene. Find the diffraction patterns of graphene from a website and define the reciprocal lattice vectors. Compare these obtained vectors with your results.

Each bright spot corresponds to a reciprocal lattice point, matching the blue points in the Python-generated plot. The structure also exhibits hexagonal symmetry, but it reflects density and directional properties. Brighter spots (center and some key points) represent lower-order reciprocal lattice points, while farther spots are higher-order with reduced intensity.

Comparison of Fig.26 with our generated plot Fig.24

- The reciprocal lattice we plotted consists of a hexagonally symmetric pattern formed by a parallelogram grid.
- Fig.26 exhibits the same structure but without the grid boundaries drawn
- The central point corresponds to the reciprocal space origin(0,0).
- The alignment and density of the points in the experimental diffraction pattern match the points in our Python-generated plot.

3. Use the lattice points and the reciprocal lattice vectors you obtained from Python to check if reciprocal lattice points can be mapped out by the set of reciprocal lattice vectors that yield plane waves with the periodicity of a given Bravais lattice.

- **Distribution of Reciprocal Lattice Points:**

In Fig.24, blue dots represent the reciprocal lattice points, while orange dashed lines indicate the unit cell of the reciprocal lattice. The red and green arrows correspond to the reciprocal lattice vectors \mathbf{b}_1 and \mathbf{b}_2 , respectively. The reciprocal lattice points generated by Python exhibit a hexagonal arrangement, which aligns with theoretical calculations and confirms the hexagonal symmetry of the reciprocal lattice.

- **Periodicity of Plane Waves:**

The mathematical form of the plane wave is:

$$\psi(r) = e^{i\mathbf{G}\cdot\mathbf{r}} \quad (29)$$

Here is the source code to generate the plane wave:

- (a) Initialize reciprocal vectors \mathbf{b}_1 and \mathbf{b}_2 . The two reciprocal lattice vectors of graphene are manually defined to describe the periodicity in reciprocal space.

```
import numpy as np
import matplotlib.pyplot as plt

# 計算倒易晶格向量
b1 = np.array([2 * np.pi / np.sqrt(3), 2 * np.pi / 3])
b2 = np.array([2 * np.pi / np.sqrt(3), -2 * np.pi / 3])
```

Figure 48: Initialize reciprocal vectors

- (b) Define a function of plane wave

```
# 定義平面波函數
def plane_wave(x, y, G):
    return np.cos(G[0] * x + G[1] * y) # 以 cos 函數表示平面波
```

Figure 49: Function of plane wave

- (c) Create a mesh grid and the corresponding plane wave (choose \mathbf{b}_1)

```
# 設定繪圖範圍
x_vals = np.linspace(-10, 10, 200)
y_vals = np.linspace(-10, 10, 200)
X, Y = np.meshgrid(x_vals, y_vals)

# 選擇一組倒易晶格向量來測試週期性
G_test = b1
wave_pattern = plane_wave(X, Y, G_test)
```

Figure 50: Create a mesh grid and the corresponding plane wave

(d) Make a plot of the plane wave

```
# 繪製平面波週期性圖案
plt.figure(figsize=(6, 6))
plt.contourf(X, Y, wave_pattern, levels=50, cmap='coolwarm')
plt.colorbar(label='Wave Amplitude')
plt.xlabel("x")
plt.ylabel("y")
plt.title("Plane Wave with Periodicity of Reciprocal Lattice")
plt.show()
```

Figure 51: Plot plane wave

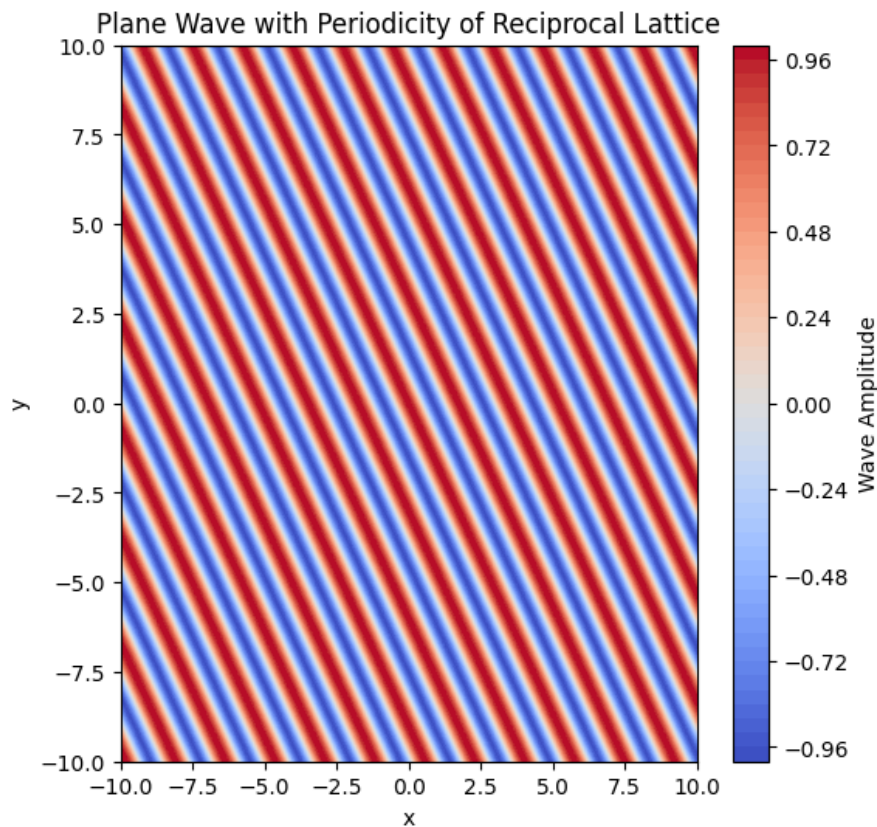


Figure 52: The plane wave corresponding to \mathbf{b}_1

The generated contour plot (Fig.52) shows that the spacing between wavefronts matches the reciprocal lattice vectors, indicating that the plane wave indeed exhibits the periodicity of the reciprocal lattice, thereby verifying the correctness of the theory.

5 Conclusion

In this experiment, we explored the representations of graphene in real and reciprocal space and analyzed the relationships between lattice vectors, reciprocal lattice vectors, and diffraction patterns. Through Python simulations, we generated both the graphene lattice structure and its corresponding reciprocal lattice. Additionally, we investigated the Moiré patterns produced by twisted bilayer graphene. We examined the relationship between their periodicity and twist angle, confirming that as the twist angle decreases, the Moiré pattern's periodicity increases.

The reciprocal lattice structure obtained from our Python simulation exhibits hexagonal symmetry, which aligns with the theoretical model. By applying the reciprocal lattice condition, we computed the reciprocal lattice vectors and verified the periodicity of the system. Furthermore, the simulated reciprocal lattice points closely match the diffraction patterns from external references. We also analyzed how reciprocal lattice vectors affect the periodicity of plane waves. Using Python, we created contour plots of wave vectors corresponding to reciprocal lattice points. The results demonstrate that these plane waves exhibit periodicity consistent with the Bravais lattice.

6 Work division

- 洪瑜：Reciprocal space, Report
- 黄巧涵：Plane wave, Report
- 洪懌平：Graphene and Moiré patterns, Report



Figure 53: 洪瑜(*left*), 黄巧涵(*middle*), and 洪懌平(*right*)

7 Source codes of these exercises

- FFT of a square pulse: https://github.com/hyp0515/exp_phy_ii/blob/main/mar18/squre_pulse.ipynb
- Generating graphene's lattice structure: https://github.com/hyp0515/exp_phy_ii/blob/main/mar18/graphene.ipynb
- Generating a twisted bilayer graphene Moiré pattern: https://github.com/hyp0515/exp_phy_ii/blob/main/mar18/moire_pattern.ipynb
- Reciprocal lattice and plane wave: https://colab.research.google.com/drive/1tj_p0CpQVOKVBAET_oL-Sx_tk9qLaxlg?usp=sharing

8 Reference

1 : <https://shorturl.at/Xephu>

2 : <https://shorturl.at/SMwrn>

3 : <https://shorturl.at/0QxmQ>