LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Determining the Number of Graphene Layers Using Atomic Force Microscopy in PeakForce Mode

Bestimmung der Anzahl von Graphenschichten mittels Rasterkraftmikroskopie im PeakForce-Modus

Bachelor Thesis

von

Ylvi Haizmann

Chair of Experimental Solid State Physics
Quantum Materials

Supervisors:

Prof. Dr. Dmitri K. Efetov
Prof. Dr. Martin Lee

07.03.2025

Abstract

In recent years two-dimensional (2D) materials have been extensively researched, due to their interesting properties, such as the tunability of the band gap, including both its magnitude and whether it is direct or indirect, depending on the number of layers of the materials. However, determining the number of layers of 2D materials can still be challenging. In this thesis, the viability of a new atomic force microscopy mode, PeakForce$^{\text{TM}}$ microscopy, is explored by measuring monolayer to hexalayer graphene flakes exfoliated on SiO_2/Si-substrates. Analyzing the PeakForce$^{\text{TM}}$ measurements showed, that the height measurements of the layers remained consistent and were able to accurately identify the number of layers up to pentalayer graphene with high accuracy, 27 out of 29 measured graphene samples could be correctly classified. Additionally, it was found that previously used techniques based on Raman spectroscopy were unable to accurately identify the number of layers for graphene flakes consisting of three or more layers. In contrast, PeakForce$^{\text{TM}}$ microscopy provided consistent height measurements, suggesting that this method could be a more reliable alternative for layer identification in 2D materials. These findings highlight the advantages of PeakForce$^{\text{TM}}$ microscopy in layer identification for graphene and suggest this method might be viable for other 2D materials.

Contents

Chapter 1

Introduction

Since the discovery of graphene in 2004, a new field of material research has opened up for the so-called two-dimensional (2D) materials [1]. The name 2D materials refers to materials that are composed of a single or a few layers of atoms, which are bonded covalently or ionically to each other, with van der Waals interactions holding the layers together in the bulk crystal, giving them a sheet-like structure [1, 2, 3]. These materials possess high mechanical strength and flexibility, as well as distinct electrical, optical, and thermal properties in the 2D limit, that differentiate them from their bulk forms [2, 4].

The number of layers of these materials can greatly influence their properties. For instance, the number of layers can be used to tune the bandgap of black phosphorus, due to its bandgap varying from 0.3 eV for the bulk and to 1.5 eV for the monolayer [5]. The band structure can also be influenced by the number of layers, for example the semiconductor MoS_2 transitions from an indirect bandgap in the bulk to a direct bandgap in the monolayer [6]. Another example of differences in the properties depending on the number of layers is graphene, in which the electrons in the monolayer behave like massless Dirac fermions, whereas in the bilayer the electrons already have a finite mass [7]. Therefore, it is important to be able to accurately quantify the number of layers, in order to harness the properties of these 2D materials.

Various techniques to identify the number of layers have already been explored. Among others, optical microscopy [8, 9, 10], Raman spectroscopy [11, 12] and atomic force microscopy in tapping mode [13, 14, 15] have been used. However, these techniques are not suitable for all 2D materials, since for example it is difficult to identify the layer of hexagonal boron nitride (hBN) with optical microscopy, due to it having a band gap larger than 5 eV and therefore not absorbing visible light and making it detectable only through the contrast caused by the change in the optical path [16]. Therefore there is a non negligible risk of misidentifying the number of layers in the few layer limit of hBN, especially when the samples are contaminated, even when placed on the optimal substrate for this method [16].

Raman spectroscopy is a fast and non-destructive technique that measures the frequency shift of the inelastically scattered photons, reflecting the lattice vibrations and electronic band structure of a material [11, 17, 18]. However, while it serves as a powerful technique to distinguish monolayer graphene from the rest [11], it is not suited for every 2D material. Using the example of hBN again, it is difficult to determine the number of layers using Raman spectroscopy, since there is one dominant Raman active mode, that is only slightly dependent on the number of layers [16, 19].

Atomic force microscopy is a technique used to create topographical maps of samples, in which a cantilever with a thin tip scans over an area and records the interactions between the tip and the sample [20, 21]. It is one of the most common methods, along with Raman spectroscopy, used to characterize graphene [15]. Tapping mode is a widely used mode of atomic force microscopy, in which the cantilever is oscillating at near resonance frequency [15, 22]. When the cantilever interacts with the sample, the amplitude of the oscillation changes. These changes in amplitude are used to generate a topographical map [23, 24]. However, there are also problems with determining the number of layers using atomic force microscopy in tapping mode, since the height measurements often overestimate the real height, due to the substrate roughness and the presence of adsorbents (buffer layer) or impurities at the interface between the substrate and the 2D material [13, 14].

Recently, PeakForce$^{\text{TM}}$ microscopy, an atomic force microscopy mode, has been tested on single layer graphene and was able to measure its height with an error of only 0.1-0.3 nm [25]. This mode relies on the tip repeatedly pressing down on the sample with a set force and generating a map of the height by measuring the deflection of the cantilever [26]. In contrast to tapping mode, where the amplitude is kept constant, PeakForce$^{\text{TM}}$ microscopy maintains a maximal force on the sample [23, 27]. This results in greater applied forces, allowing for compression of the buffer layer between the substrate and the 2D material [25].

In this thesis, the use of PeakForce$^{\text{TM}}$ microscopy is explored to quantify the number of layers of graphene. This new method is benchmarked on graphene, since the layers of graphene can already be reliably determined using optical microscopy and Raman spectroscopy. This allowes for a direct comparison of PeakForce$^{\text{TM}}$ microscopy with other characterization methods, such as optical spectroscopy, Raman spectroscopy, and tapping mode, to evaluate their accuracy.

The results confirm PeakForce$^{\text{TM}}$ microscopy as a reliable method for determining the number of layers of graphene with high accuracy, especially when measuring multiple different locations on a flake. While the analysis of the Raman spectroscopy in this thesis can only identify monolayer and bilayer graphene flakes, it is unable to distinguish between flakes with three or more layers, making it unsuitable for multilayer characterization.

In Chapter 2, a general introduction to graphene is provided, along with the theoretical background on optical contrast, Raman spectroscopy and atomic force microscopy, including its different modes. In Chapter 3, the exfoliation process of graphene is first described. Additionally, optical contrast, Raman spectroscopy, tapping mode, and PeakForce^{TM} are explained, as well as the methods used to identify the number of layers of graphene. In Chapter 4, the analysis of the layer identification of Raman spectroscopy, tapping mode and PeakForce^{TM} is given, comparing their results to the number of layers determined by optical contrast and assessing their accuracy.

Chapter 2

Theoretical Background

In this thesis, characterization of the number of layers with PeakForce$^{\mathrm{TM}}$ microscopy is performed. Graphene is selected as the analysis object, since there are already reliable methods, such as optical contrast [8, 9, 10] and Raman spectroscopy [11, 12], to determine the number of layers of graphene, allowing for an assessment of the accuracy of the PeakForce$^{\mathrm{TM}}$ measurements.

To this end, this Chapter provides a general introduction to graphene, with a focus on the relevant parameters for the experiments conducted in this thesis. Additionally, the theoretical background of the methods used to characterize of the number of graphene layers is discussed.

2.1 Graphene

Graphene is a single atomic layer of graphite, composed of a hexagonal grid of carbon atoms [12]. Each carbon atom possesses six electrons, which in the ground state occupy the $1\mathrm{s}^2$, $2\mathrm{s}^2$, $2\mathrm{p}_x^2$ and $2\mathrm{p}_y^2$ orbitals. When bonding with other atoms, one of the 2s electrons is promoted to the empty $2\mathrm{p}_z$ orbital, enabling the formation of hybrid orbitals [28]. Of the three p-orbitals, only two get hybridized forming three sp^2-orbitals, that arrange themselves to be 120° apart in a 2D structure. These orbitals form the in-plane σ-bonds, forming covalent bonds with three neighboring atoms, which establish the honeycomb structure. The remaining 2p electron stays in the $2\mathrm{p}_z$-orbital, which is oriented perpendicular to the sp^2-orbitals. In graphite this orbital binds the individual sheets to the neighboring ones, that are in the planes above and below it, by the van der Waals force to form the interplane π-bond [28, 29].

The binding energy of the π-bond is significantly lower than that of the σ-bonds, making graphite layers easy to shear [28]. The overlap between the $2\mathrm{p}_z$ orbitals of adjacent carbon atoms causes the electrons to delocalize and form a valence band, called π band, which is occupied by the electron and an empty conduction band, called π^* band [29, 30].

The graphene lattice consists of a triangular lattice, with a basis of two carbon atoms A and B, in each rhomboid unit cell (Fig. 2.1(a)). The vectors of the unit cell are

$$\vec{a}_1 = \begin{pmatrix} \sqrt{3}a/2 \\ a/2 \end{pmatrix} \qquad \vec{a}_2 = \begin{pmatrix} \sqrt{3}a/2 \\ -a/2 \end{pmatrix}, \tag{2.1}$$

where $a = \sqrt{3}a$ with $a = 0.142$ nm being the distance between two atoms in-plane. The basis vectors of the reciprocal lattice are

$$\vec{b}_1 = \begin{pmatrix} 2\pi/\sqrt{3}a \\ 2\pi/a \end{pmatrix} \qquad \vec{b}_2 = \begin{pmatrix} 2\pi/\sqrt{3}a \\ -2\pi/a \end{pmatrix}, \tag{2.2}$$

with a reciprocal lattice constant of $4\pi/\sqrt{3}a$ [28, 29].

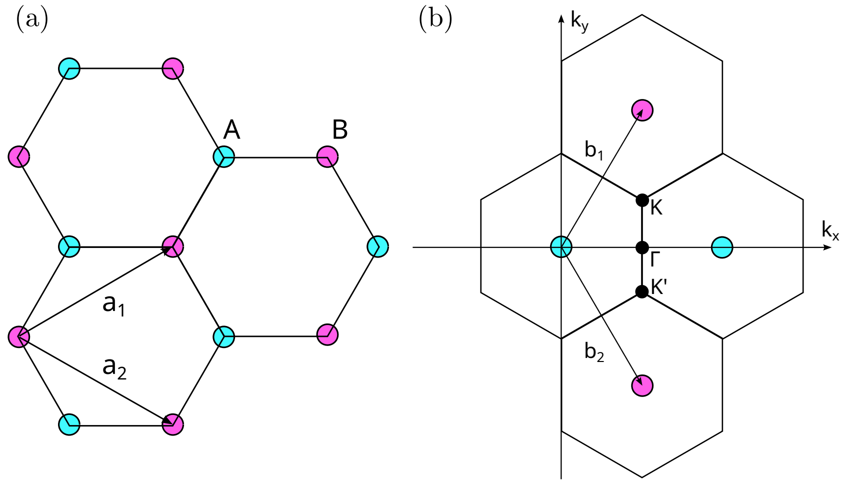


Figure 2.1: (a) Hexagonal lattice of graphene in real space with a_1 and a_2 as the vectors of the unit cell. (b) Reciprocal graphene lattice with b_1 and b_2 as the basis vectors, adapted from Ref. [31]

The electronic bandstructure of graphene can be approximated using the tight-binding Hamiltonian, considering nearest and next nearest neighbours hopping and setting $\hbar = 1$

$$\hat{H} = -\gamma_0 \sum_{\langle i,j\rangle,\sigma} \left(a^\dagger_{\sigma,i} b_{\sigma,j} + h.c. \right) - \gamma_0' \sum_{\langle\langle i,j\rangle\rangle,\sigma} \left(a^\dagger_{\sigma,i} a_{\sigma,j} + b^\dagger_{\sigma,i} b_{\sigma,j} + h.c. \right), \tag{2.3}$$

where $a_{\sigma,i}$ ($a^\dagger_{\sigma,i}$) is the annihilation (creation) operator for electrons on site $\mathbf{R}_i$ on the sublattice A, $b_{\sigma,j}$ ($b^\dagger_{\sigma,j}$) is the annihilation (creation) operator for electrons on site $\mathbf{R}_j$ on the sublattice B. The spin is denoted by $\sigma = \uparrow, \downarrow$. $\gamma_0 \approx$2.8 eV is the required energy for nearest neighbour hopping between different sublattices and γ_0' is the energy required for next nearest neighbour hopping within the same sublattice [31].

Solving this Hamiltonian, the eigenvalues as a function of the wave vector ($\mathbf{k}$) are

$$E_{\pm}(\mathbf{k}) = \pm\gamma_0\sqrt{3+f(\mathbf{k})} - \gamma_0' f(\mathbf{k}), \tag{2.4}$$

where

$$f(\mathbf{k}) = 2\cos\left(\sqrt{3}k_y a\right) + 4\cos\left(\frac{\sqrt{3}}{2}k_y a\right)\cos\left(\frac{3}{2}k_x a\right), \tag{2.5}$$

The positive eigenvalue corresponds to the upper π-band (conduction band), while the negative eigenvalue corresponds to the lower π^*-band (valence band) [12, 31]. These bands are symmetric around zero energy for $\gamma_0' = 0$, Fig. 2.2 (a) shows an asymmetric band structure for $\gamma_0' \neq 0$ [31].

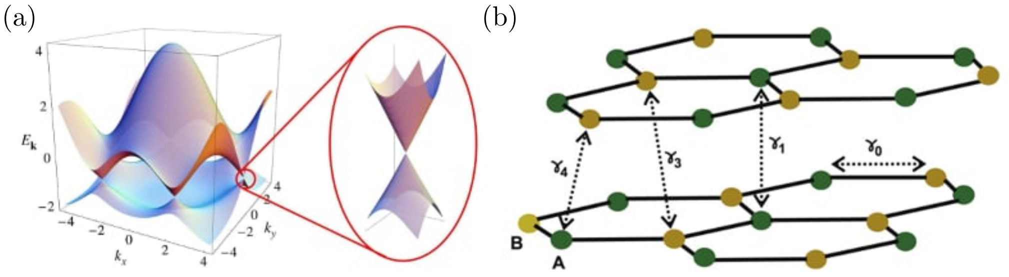


Figure 2.2: (a) Bandstructure of monolayer graphene with t =2.7 eV and $t' = 0.2t$, adapted from Ref. [31] (b) Lattice of bilayer graphene with the nearest neighbour hopping possibilities, adapted from Ref. [12]

The tight binding model can be expanded by considering interlayer interactions to describe the bandstructure of multilayer graphene [31]. The Hamiltonian for bilayer graphene, assuming half of the atoms in one layer align directly above the atoms of the lower layer [32], is given by

$$\begin{aligned}\hat{H} = &-\gamma_0 \sum_{\langle i,j\rangle,m,\sigma}\left(a^\dagger_{m,\sigma,i}b_{m,\sigma,j} + h.c.\right)\\ &-\gamma_1\sum_{j,\sigma}\left(a^\dagger_{1,\sigma,j}a_{2,\sigma,j} + h.c.\right)\\ &-\gamma_3\sum_{j,\sigma}\left(a^\dagger_{1,\sigma,j}b_{2,\sigma,j} + a^\dagger_{2,\sigma,j}b_{1,\sigma,j} + h.c.\right)\\ &-\gamma_4\sum_{j,\sigma}\left(b^\dagger_{1,\sigma,j}b_{2,\sigma,j} + h.c.\right),\end{aligned} \tag{2.6}$$

where the first sum represents in-plane hopping within each layer $m = 1, 2$ and the other sums account for the other hopping possibilities, see Fig. 2.2 (b) with $\gamma_1 \approx$ 0.4 eV, $\gamma_3 \approx$ 0.3 eV and $\gamma_4 \approx$ -0.04 eV being the hopping energies [31]. This results in bilayer graphene having two conduction and valence bands [33]. Adding more graphene layers leads to further band splitting, making it possible to differentiate the number of layers using Raman, as further discussed in Chapter 2.3 [12].

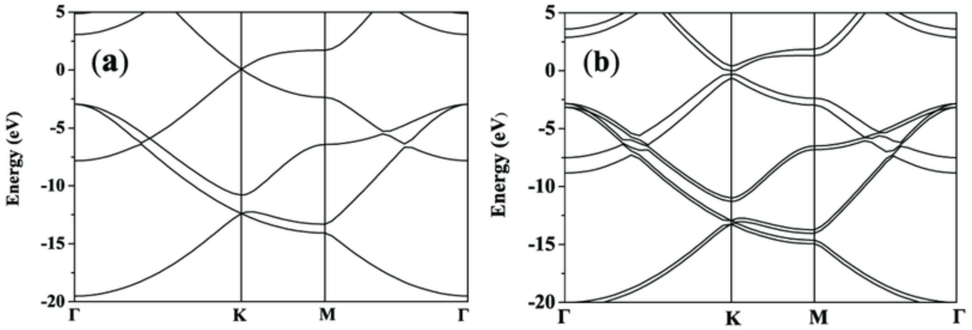


Figure 2.3: Band structure diagram of monolayer (a) and bilayer (b) graphene, taken from Ref. [34]

2.2 Optical Contrast

Optical contrast is a fast and nondestructive method to reliably determine the thickness of graphene flakes [8]. This method is based on Fresnel's law, which describes how light behaves at an interface between two materials with different refractive indices [9, 35]. The reflected light intensity depends on the wavelength of the light, as well as on the refractive indices and thicknesses of the different materials, which makes it possible to distinguish between the different layer numbers of graphene. The formula for the reflected light intensity is

$$\begin{aligned} I(n_1) = \Bigg| & \left(\frac{n_0-n_1}{n_0+n_1}e^{i(\Phi_1+\Phi_2)} + \frac{n_1-n_2}{n_1+n_2}e^{-i(\Phi_1-\Phi_2)} + \frac{n_2-n_3}{n_2+n_3}e^{-i(\Phi_1+\Phi_2)}\right) \\ & \Bigg(e^{i(\Phi_1+\Phi_2)} + \frac{n_0-n_1}{n_0+n_1}\frac{n_1-n_2}{n_1+n_2}e^{-i(\Phi_1-\Phi_2)} \\ & + \frac{n_0-n_1}{n_0+n_1}\frac{n_2-n_3}{n_2+n_3}e^{-i(\Phi_1+\Phi_2)} + \frac{n_1-n_2}{n_1+n_2}\frac{n_2-n_3}{n_2+n_3}e^{i(\Phi_1+\Phi_2)}\Bigg)^{-1}\Bigg|^2, \end{aligned} \tag{2.7}$$

with $n_0 = 1$ being the refractive index of air, n_1 being the refractive index of graphene, n_2 being the refractive index of silicon dioxide (SiO_2) and n_3 being the refractive index of silicon (Si) [8, 9]. $\Phi_1 = 2\pi n_1 d_1/\lambda$ and $\Phi_2 = 2\pi n_2 d_2/\lambda$ describe phase differences, due to the interference of reflected light which stems from the optical path difference, where d_1 is the thickness of the graphene, d_2 is the thickness of the SiO_2 and λ is the wavelength, upon which all the refractive indexes depend on. The thickness of the Si does not appear in the formula since Si can be considered semi-infinite [8, 9]. see the schematic diagram in Fig. 2.4.

By comparing the calculated intensities of the reflected light with ($n_1(\lambda)$) and without ($n_1 = 1$) the graphene monolayer on the substrate, the contrast C can be determined using

$$C = \frac{I(n_1 = 1) - I(n_1(\lambda))}{I(n_1 = 1)}. \tag{2.8}$$

With this, it was found, that for graphene when using a substrate with a SiO_2-layer with a thickness of approximately 280 nm the highest contrast is visible in green light [9].

The contrast of graphene was observed to increase almost linearly for graphene monolayers up to nine layers, making it possible to differentiate between the different number of layers of graphene by analyzing the change in green value from the bare substrate to the sample [10]. This method needs to be configured in each laboratory, since the contrast calibration varies from one equipment to another [8].

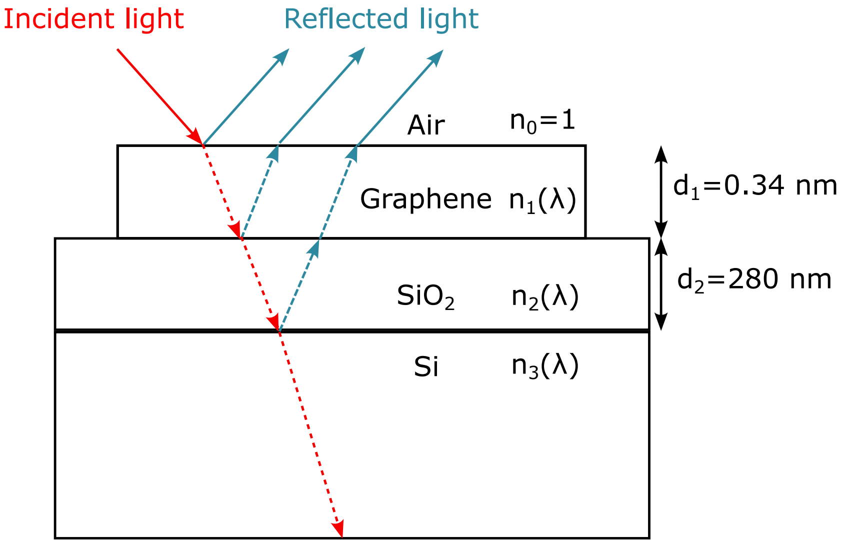


Figure 2.4: Schematic diagram of light reflection and transmission for monolayer graphene on a 280 nm SiO_2/Si substrate, adapted from Ref. [9]

2.3 Raman Spectroscopy

Raman spectroscopy relies on the inelastic scattering of high-energy photons [36]. It measures the frequency shift of scattered photons, which occurs when an incoming photon transfers energy to or from a phonon, a vibrational mode of a molecule [36, 37]. Since crystals have energy levels that are unique to them, they can be identified by their characteristic Raman spectra [38, 39].

Raman spectroscopy has also emerged as a tool to study the vibrational modes of 2D materials. In particular, Raman spectroscopy has proven to be effective in identifying the number of layers of graphene and many other 2D materials due to the changes in the electronic and phononic band structures as a function of number of layers [12, 40]. This makes it possible to identify the number of layers of graphene by the position, shape and intensity of the peaks changing with the number of layers. The most prominent peaks in the Raman spectra of defect-free graphene are the G and the second order D band, also called the 2D band, see Fig. 2.5 [11].

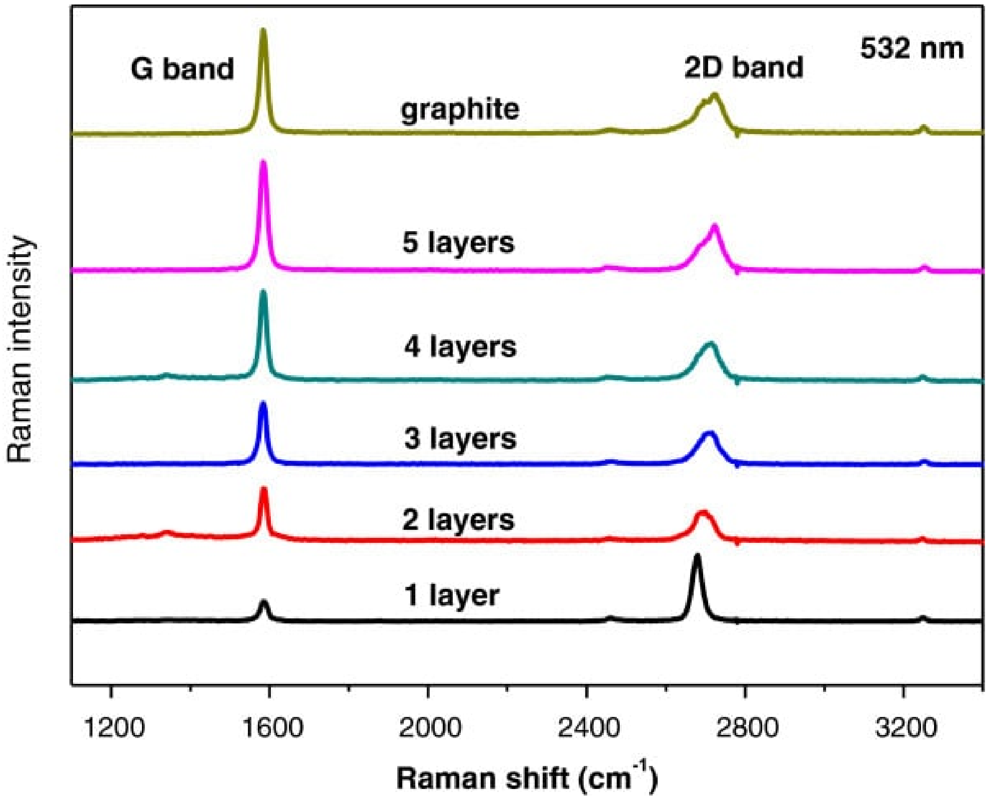


Figure 2.5: Raman spectra of graphene measured with a 532 nm laser for mono-, bi-, tri-, tetra-, pentalayer and graphite. Taken from Ref. [41].

The G band, located at around 1580 cm^{-1} when using a 2.33 eV laser, arises from in-plane stretching vibrations associated with optical phonons at the Brillouin zone center (Γ point) and is the only first order Raman scattering process in graphene [11, 12]. During this scattering process a photon interacts with an electron and the electron is promoted to a higher energy state, creating an electron-hole pair. A phonon is then emitted due to an electron-phonon interaction, after which the electron recombines with the electron hole, emitting a photon [12, 42, 43]. The intensity of the G peak increases with the number of graphene layers, since the photon interactions with the in-plane vibrations, see Fig. 2.6 (a), increase [11].

The 2D band, found at around 2700 cm^{-1} using a 2.33 eV laser, is caused by double resonance processes. This process involves four successive transitions: First, an electron in the valence band absorbs a photon and is excited to the conduction band, creating an electron-hole pair. The electron then undergoes inelastic scattering from point K to point K′ in the Brillouin zone due to a phonon before being

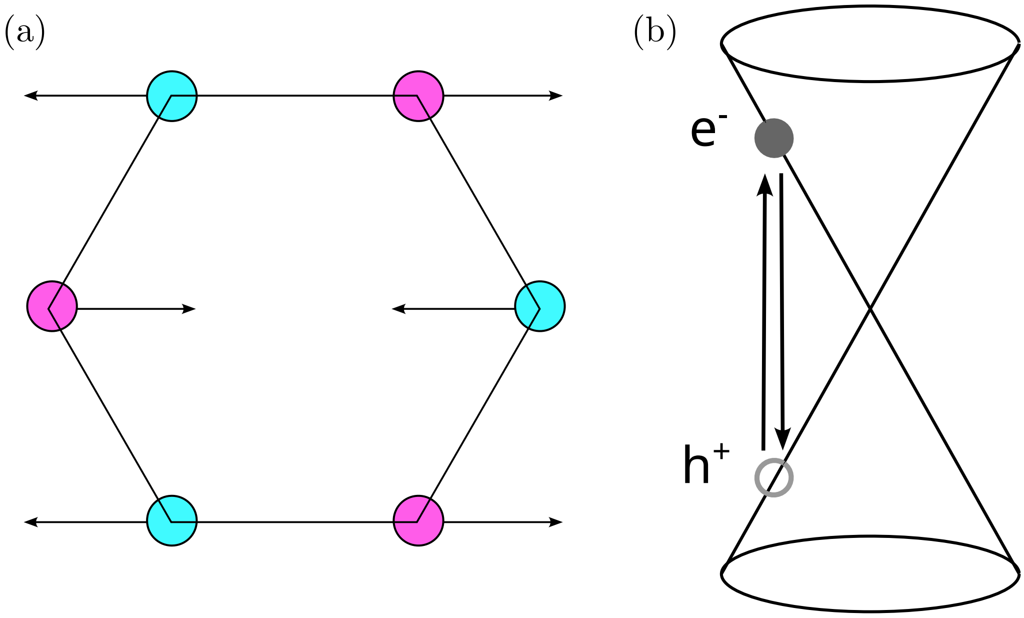


Figure 2.6: (a) In-Plane Vibrations of the G band, adapted from Ref. [44] (b) Raman Process of the G Band, adapted from Ref. [12].

scattered back to point K by another phonon. Finally, the electron recombines with the hole in the valence band by emitting a photon, as shown in Fig. 2.7 (a) [11, 12].

In monolayer graphene, the 2D peak is symmetric and can be accurately fitted with a single Lorentzian peak, since in the monolayer there is only one possible double resonance scattering process. The 2D peak can be explained by triple resonance processes, where instead of the electron scattering back, the electron-hole is scattered to K′ as well, and is then recombined with the electron by photon emission there, as shown in Fig. 2.7 (b) [12].

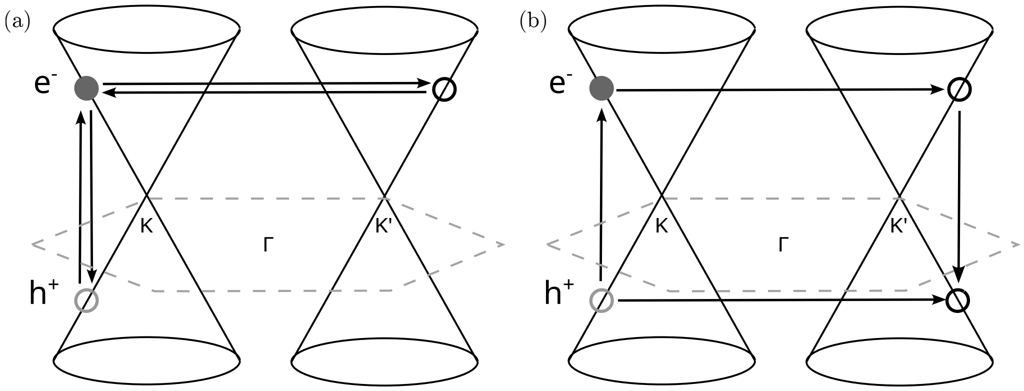


Figure 2.7: (a) Double Resonance Process of the 2D-Band (b) Triple Resonance Process of the 2D-Band in Monolayer Graphene, Figure adapted from Ref. [12].

In bilayer graphene, interlayer interactions lead to the splitting of both the electron and phonon bands into two components. The phonon bands in the two layers exhibit a vibration, that is in-phase for one component and out-of-phase for the other. Similarly, the electron bands split into symmetric and anti-symmetric components, resulting in two conduction bands (π_1 and π_2) and two valence bands (π_1^* and π_2^*), corresponding to the lower and upper bands of each type, as shown in Fig. 2.8 (f) [12].

In these bands, as in the monolayer, electron-hole pairs can form between the valence and conduction bands, but now, depending on the polarisation of the absorbed photon, the electron-hole pair can form between $\pi_1 \rightleftarrows \pi_1^*$ (or $\pi_2 \rightleftarrows \pi_2^*$) and $\pi_1 \rightleftarrows \pi_2^*$ (or $\pi_2 \rightleftarrows \pi_1^*$). The electrons, which can occupy two excited states, can then be scattered by phonons with two different symmetries. Depending on the symmetry of the phonon, the electron is scattered from $\pi_1^* \rightarrow \pi_1^*$ (or $\pi_2^* \rightarrow \pi_2^*$) or from $\pi_1^* \rightarrow \pi_2^*$ (or $\pi_2^* \rightarrow \pi_1^*$). Due to this, there are four possible double resonance processes, each giving rise to a Raman peak [12].

In the trilayer, there are six carbon atoms in the unit cell, which have five possibilities for creating an electron-hole pair. These electrons can be scattered by three different optical phonon branches, giving rise to fifteen possible double resonance processes. Even though there are fifteen possible transitions, it has been shown, that the minimum number of Lorentzians necessary to fit the Raman peak correctly are only six, as the transitions are very close to each other [12].

The Raman spectra of tetralayer graphene marks the transition, from where the Raman spectra of graphene begins to resemble that of highly oriented pyrolytic graphite. The 2D-peaks from tetralayers up can be fitted accurately with just three peaks [12].

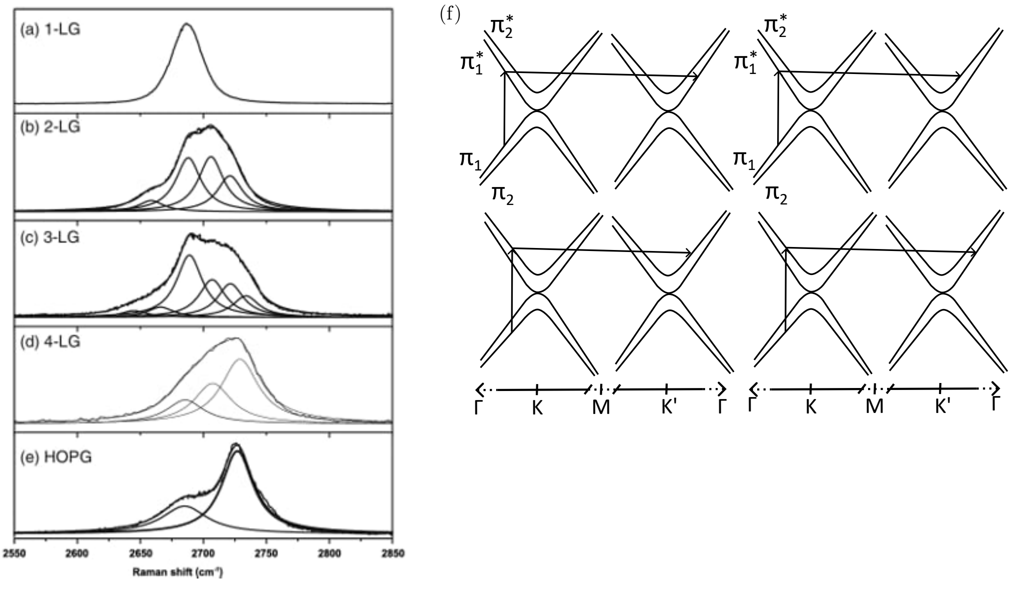


Figure 2.8: (a)-(e) 2D-peak of mono- (a), bi- (b), tri- (c), tetra-layer (d) graphene and highly oriented pyrolytic graphite (e) with Lorentzian-peaks, taken from Ref. [12]
(f) The four possible transitions for bilayer graphene, adapted from Ref. [12].

Due to this, the position of the 2D-peak [40, 45, 46], the full width at half maximum (FWHM) of the 2D-peak [11, 45] and the ratio of the integrated intensity of the G-peak to the 2D-peak can be used to determine the number of layers [40, 47].

2.4 Spring Constant

As the spring constant plays a vital role in atomic force microscopy, a short introduction is given in this section.

The spring constant quantifies a material's resistance to deformation under an applied force. This relationship is described by Hooke's law:

$$\mathbf{F} = k\mathbf{x}, \tag{2.9}$$

where $\mathbf{F}$ is the force applied to the material, k is the spring constant and $\mathbf{x}$ is the amount it is deformed, as seen in Fig. 2.9 [48].

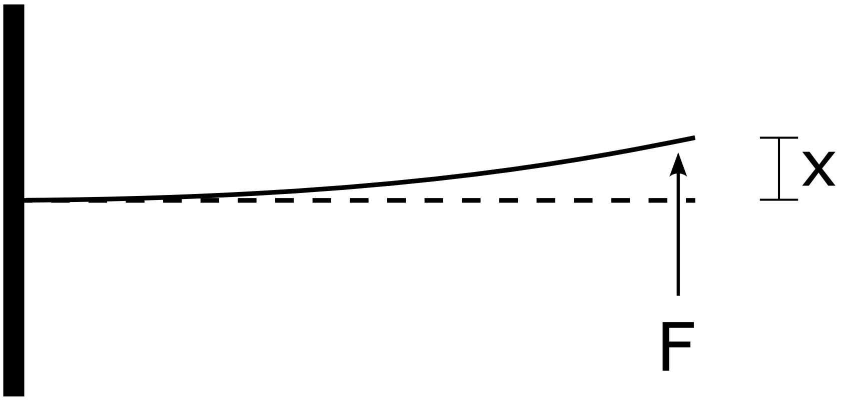


Figure 2.9: Schematic Illustration of a Bent Cantilever, adapted from Ref. [49]

The spring constant depends on the geometry of the object, whereas Young's modulus is an intrinsic material property independent of shape or size. Young's modulus can be related to the spring constant, when the cross section and the length of the material are known, using the formula:

$$E = k\frac{L}{A}, \tag{2.10}$$

where E is the Young's modulus, L is the length of the material and A the cross section [48].

Since the spring constant affects the amount by which the cantilever, see Fig. 2.10, deforms under a specific force during atomic force measurements, it is necessary to determine it accurately to be able to extract the precise height measurements [50].

In the case of a rectangular cantilever, the spring constant can be calculated with:

$$k = \frac{EWT^3}{4L^3}, \tag{2.11}$$

where W is the width, T is the thickness and L is the length of the cantilever [51]. However, the width, length and especially the thickness of the cantilever are hard to accurately determine and have an uncertainty of ±3%, ±3% and±5%, adding up to an uncertainty of the spring constant of ±20%, where the uncertainty of the thickness is the main contributor [52]. Due to the large uncertainty, it is desirable to have an alternative method to determine the spring constant, which does not depend on the geometry of the cantilever.

The resonance frequency of the cantilever is dependent on the spring constant and, in a vacuum, can be calculated by

$$f_0 = \frac{1}{2\pi}\left(\frac{k}{m_0}\right)^{\frac{1}{2}}, \tag{2.12}$$

where m_0 is the mass of the cantilever [51]. Thus, the spring constant can also be determined by measuring the resonance frequency, as explained in Chapter 2.5.4.

2.5 Atomic Force Microscopy

The atomic force microscopy (AFM) is used for taking force measurements and topographical images of the surfaces of samples on an atomic scale. It was invented by Binnig et al. in 1986 [21].

Imaging is achieved through the use of a cantilever tip scanning a sample and measuring the interaction between the tip and the sample [20]. These interactions are determined by measuring the displacement of the free end of the cantilever [26]. This is accomplished by reflecting a laser beam onto the cantilever and directing it into a segmented photodetector [20]. Before beginning the measurement, the photodetector is adjusted to ensure that the voltage across the four segments is equal. As the cantilever is deflected, the reflection angle of the laser changes, and a voltage proportional to the deflection is measured [53].

The deflection can be calculated using the following formula:

$$q \simeq \Delta s \frac{L}{D}, \tag{2.13}$$

where q is the displacement of the cantilever, Δs is the distance the laser spot has moved on the photodiode, L is the length of the cantilever and D is the distance of the cantilever to the photodiode [53]. To scan the sample, the cantilever is moved

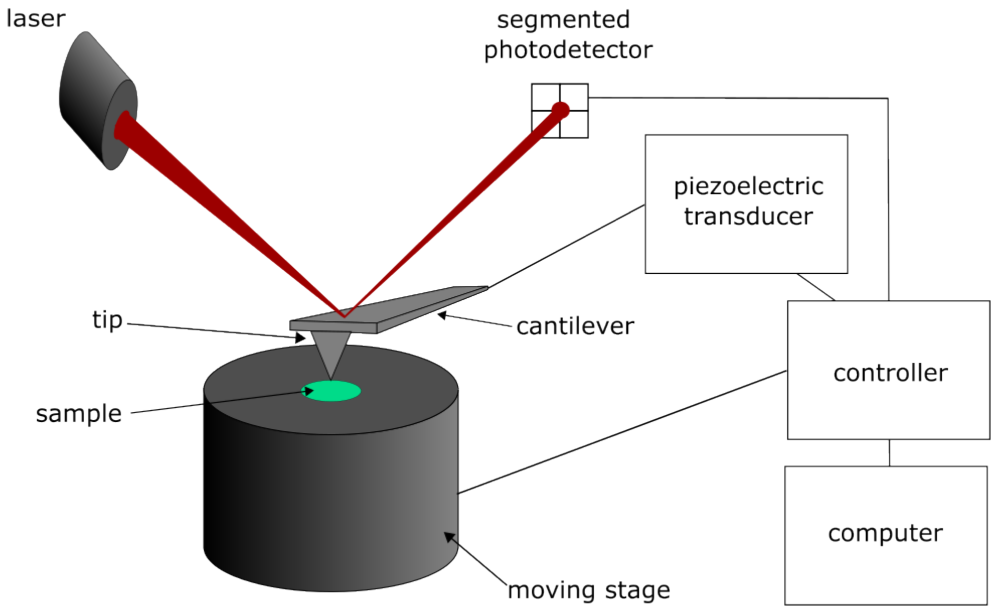


Figure 2.10: Schematic Design of an AFM. Adapted from Ref. [21].
The AFM consists of a flexible cantilever, which has a sharp tip at one end and is connected to a piezoelectric transducer (piezo), which is responsible for moving the cantilever in the xyz directions, at the other end. A laser is focused on the back of the cantilever and reflected onto a four-segment photodetector [21]. The sample is placed on a moving stage, which is used to position the sample and adjust its location before the measurement begins.

by a piezoelectric transducer (piezo) in a raster pattern [20].

While scanning, the interaction forces between the tip and the sample can be described by the Lennard-Jones potential:

$$F(r) = -\frac{A}{r^6} + \frac{B}{r^{12}}, \tag{2.14}$$

where r is the distance between the tip and the sample, and A and B are parameters, which can be derived from experimental data. The first term acts as an attractive force on the cantilever and represents the van der Waals interaction, originating from the dipole-dipole interactions. It dominates at long distances. The second term represents the Pauli repulsion, caused by the overlap of the electron clouds, dominating at short distances [23, 54].

There are many different AFM modes, which can be split into three different categories each designed to probe different tip-sample interactions along the force curve. If the tip-sample interactions remain mainly in the repulsive regime, keeping the deflection of the cantilever constant during the measurements, the mode is referred to as contact mode. When measurements are performed in the attractive regime, oscillating the cantilever near its resonance frequency in a sinusoidal motion, it is called non-contact mode. Modes operating in the regime in between are

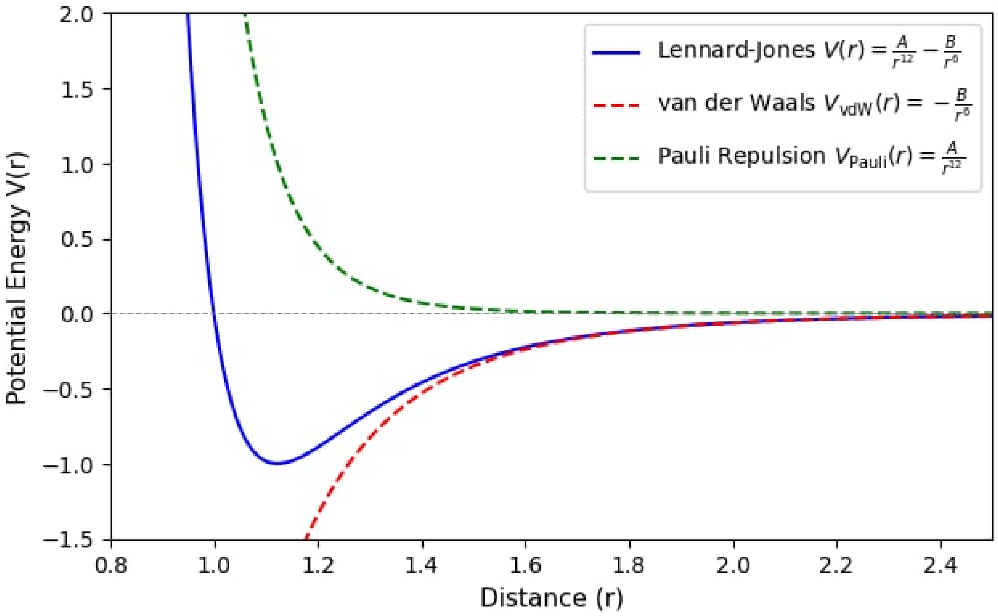


Figure 2.11: Lennard-Jones Potential (solid blue line) is a combination of van der Waals attraction (dashed red line) and Pauli repulsion (dashed green line) potentials.

called intermittent-contact or tapping mode, where the tip only briefly touches the sample [53].

In the thesis, the focus is on intermittent-contact modes, specifically tapping mode and PeakForce^{TM} tapping, as contact mode can damage the sample [55] and non-contact mode is mostly preferable for soft samples [53].

2.5.1 Contact Mode

Contact mode mainly operates in the repulsive regime of the force curve [24, 56]. In contact mode, the force between the tip and the sample is maintained constant by a feedback loop, which adjusts the height of the cantilever using a piezo [24, 26].

When starting the measurement, the tip is lowered until it is in contact with the sample, and the required force is reached. While scanning the sample in a raster motion, the deflection of the cantilever is continuously monitored and compared to the deflection corresponding to the required force. This feedback loop is then used to adjust the cantilever position to retain the required force. A topographical map of the sample is created by using the height of the cantilever and the position of the tip on the plane [24, 56]. However, a significant disadvantage of this mode is the shear force, caused by laterally dragging the tip across the sample, which can damage both the sample and tip, and distort the image [24, 56].

2.5.2 Tapping Mode

In tapping mode, the cantilever is vibrated at or near its resonance frequency by the shaker piezo on which the cantilever is mounted, due to this, the cantilever oscillates vertically [22]. The resonance frequency of the cantilever is found by vibrating the piezo over a range of frequencies, while the cantilever is not interacting with the sample and at the resonance frequency, the oscillation amplitude of the cantilever peaks. As the cantilever is moved closer to the sample, the tip starts to touch the sample briefly at the lower points of the oscillation. When the tip contacts the sample, the oscillation amplitude decreases. During the measurement, a feedback loop maintains a constant amplitude by comparing changes in amplitude to a reference amplitude and generating an error signal. This error signal is then transmitted to the piezo, which changes the distance between the tip and the sample to restore the reference amplitude. The height of the cantilever and its lateral position are used to create a topographical map [23, 24].

2.5.3 PeakForce$^{\text{TM}}$ Tapping

In PeakForce$^{\text{TM}}$ Tapping mode, the cantilever oscillates at a frequency that is considerably lower than its resonance frequency. The driving signal, which manipulates the cantilever's vertical position, is a sinusoidal wave, which is preferred to a triangular wave to avoid stimulating an unwanted oscillation, where resonance could be inadvertently triggered. As it gets closer to the sample through this motion, the cantilever gets attracted by the van der Waals force. After it makes contact with the sample the Pauli repulsion takes over and the cantilever deflects upwards [26]. When retracting, the deflection curve shows that the tip adheres to the sample before separating from the sample and then transitions into a ringing motion [26, 27].

The maximum interaction force on the cantilever is measured using a synchronization algorithm, in which the synchronization window (Sync distance in Figure 2.12) is set to half of the sine wave's period. This method allows separating the interaction region from the non-interaction region and, with that, to filter out the ringing signal, even if it exceeds the setpoint, which makes it possible to measure with a low setpoint. Additionally, by averaging over the non-interaction region to establish a baseline, the system can reengage before the ringing signal stops, speeding up the measurement process [27].

By measuring the maximum interaction force at each point on the surface, a topographical map is created, where the maximum interaction force correlates to surface height variations [27].

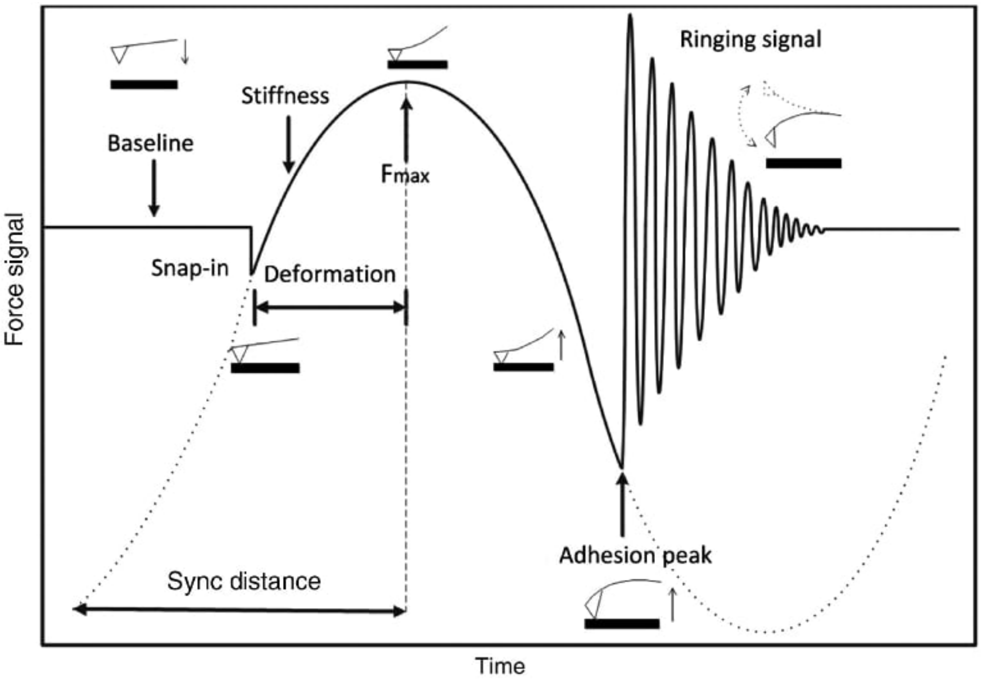


Figure 2.12: Illustration of the operational principle of the PeakForce^{TM} tapping mode, taken from Ref. [27].
The dashed line shows the driving signal of the cantilever. The thick black line shows the deflection of the cantilever.

2.5.4 Thermal Tune

For accurate AFM force measurements, obtaining a precise estimation of the spring constant is needed [57]. The spring constants provided by manufacturers, such as the μmasch HQ:NSC18/Al BS cantilever used for all the following PeakForce^{TM} measurements, often include a large variation range [58]. To achieve a more accurate determination of both the spring constant and the deflection sensitivity, the “thermal tune“ function of the AFM software can be utilized [57].

By using this function, the cantilever is kept at a position, where it does not interact with the sample. During this process, the deflection signal is recorded over a time interval, while the cantilever is not oscillated by the piezo. The movement of the cantilever results from the air molecules, moving with the Brownian motion, colliding with it [59, 60].

According to the equipartition theorem,

$$\langle\frac{1}{2}k_{spr}z^2\rangle = \frac{1}{2}k_BT, \tag{2.15}$$

where k_{spr} is the spring constant of the cantilever, z is the deflection of the cantilever k_B is the Boltzmann constant and T is the temperature [59, 60].

By Fourier transforming the deflection signal, the Power Spectral Density (PSD) is obtained in the frequency domain [60]. By fitting the resonance peak with a

Lorentzian function

$$A(v) = A_0 + \frac{C_1}{(v - v_0)^2 + C_2}, \tag{2.16}$$

with A_0 as the baseline amplitude, v_0 as the position of the resonance peak and C_1 and C_2 are Lorentzian fit parameters and integrating the area under the resonance frequency using the Lorentzian fit, the power of the fluctuations of the cantilever is obtained [57, 61]. The power spectrum corresponds to the mean square of the displacement of the cantilever during the measurement [50, 60]. With this the equation can be reformulated to

$$k_{spr} = \frac{k_B T}{\langle z^2 \rangle} = \frac{k_B T}{P}, \tag{2.17}$$

where P is the thermal fluctuation power, to calculate the spring constant [50].

Chapter 3

Methods

In this Chapter, the exfoliation process and the devices, that are used, are described, as well as, how the data, that is extracted, is used to identify the number of layers of graphene. In total the following four methods are employed: optical contrast, Raman spectroscopy, tapping mode AFM and PeakForce^{TM} microscopy, to be able to assess the accuracy of the PeakForce^{TM} measurements in chapter 4.

3.1 Exfoliation

For the fabrication of the mono- and few-layer graphene flakes, mechanical exfoliation as described in [62] is performed.

As a substrate, Si with a 285 nm thick SiO_2 layer on top of it is used, since this SiO_2 thickness provides an optimal contrast to detect graphene flakes on it [9]. This substrate is cleaned by first blowing it with compressed air to remove any large particles. The substrate is then placed in an acetone bath and ultrasonically cleaned using an EMAG Ultrasonic cleaner Emmi-H22. Removing the substrate from the acetone bath, it is dipped in 2-propanol and dried with compressed air to remove any acetone residue. Afterwards, the substrate is placed into a Femto low-pressure plasma system from Diener electronic GmbH & co. KG and cleaned with oxygen-plasma for 10 minutes, to eliminate the organic adsorbates and to improve adhesion of graphene to the substrate [62].

By sticking the graphite onto the adhesive tape and then removing the graphite from the tape, a thick graphite flake is left adhered to the tape. The tape is repeatedly folded and stuck onto thicker parts of the graphite and pealed apart to split the graphite again. This is repeated until there were many translucent gray flakes visible on the tape.

The tape with the graphene is then pressed onto the SiO_2/Si substrate and heated on a heating plate with a glass layer between the heating plate and the substrate at

115°C for 2 minutes. Heating the sample forces the gas molecules between the SiO_2 and the graphene to escape, due to the increasing pressure [62]. When allowing the sample to cool down, the decrease in pressure establishes a more uniform contact between the substrate and the graphene. The tape is then slowly peeled off the cooled-down sample.

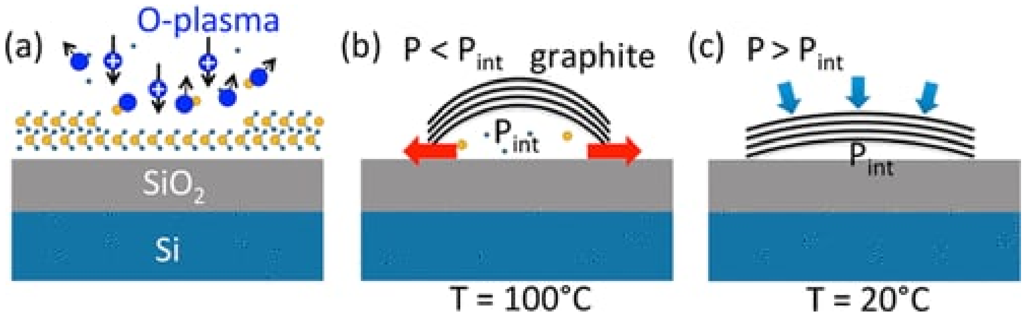


Figure 3.1: (a) Cleaning process of SiO_2/Si substrate with oxygen plasma (b) Gas molecules between the substrate and the graphene forced to escape, due to increased pressure by heating (c) Pressure decrease by cooling the sample establishes a more uniform contact between the substrate and the graphene, taken from Ref. [62]

3.2 Optical Spectroscopy

The exfoliated graphene flakes are then first characterized using optical contrast. For this, a Nikon Eclipse LV150NA microscope and its software, NIS-Elements, are used. The number of layers of the graphene flakes is identified under white light by analyzing the change in the intensity of the value of green light from the bare substrate to the part covered by the graphene flake and comparing the change to an internal calibration.

3.3 Raman Spectroscopy

Further the graphene flakes are analyzed using Raman spectroscopy (Raman). It is carried out in an argon-filled glove-box with a WITec alpha300 R Raman Imaging Microscope using a 532 nm laser and a 100x objective lens with a numerical aperture of 0.9. The laser power is set to 10.1 mW and the integration time to 0.5 s.

In total, 30 graphene flakes are measured with Raman spectroscopy. Of these 30 flakes, six are monolayers, twelve bilayers, four trilayers, four tetralayers, two pentalayers and two hexalayers. For this analysis the number of layers determined by optical contrast is assumed to be correct.

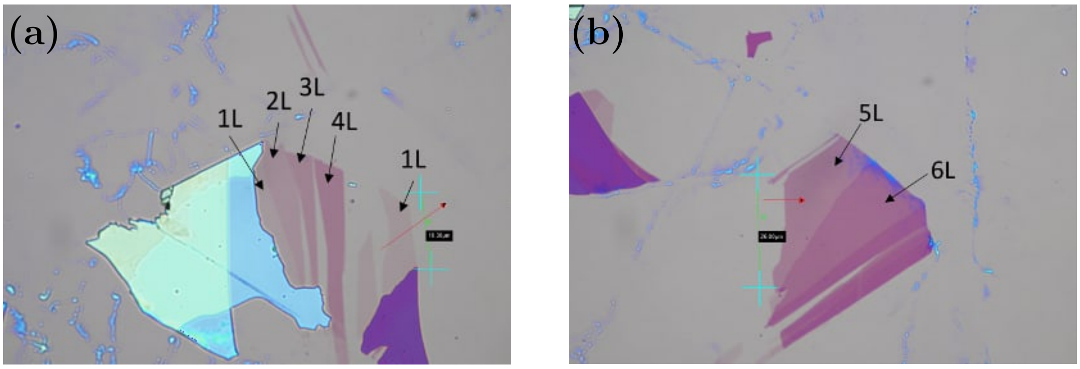


Figure 3.2: Images of graphene flakes showing (a) monolayer (1L), bilayer (2L), trilayer (3L) and tetralayer (4L) and (b) pentalayer (5L) and hexalayer (6L) using an optical microscope, with a green scale bar of (a) 18.35 μm and (b) 26.00 μm.

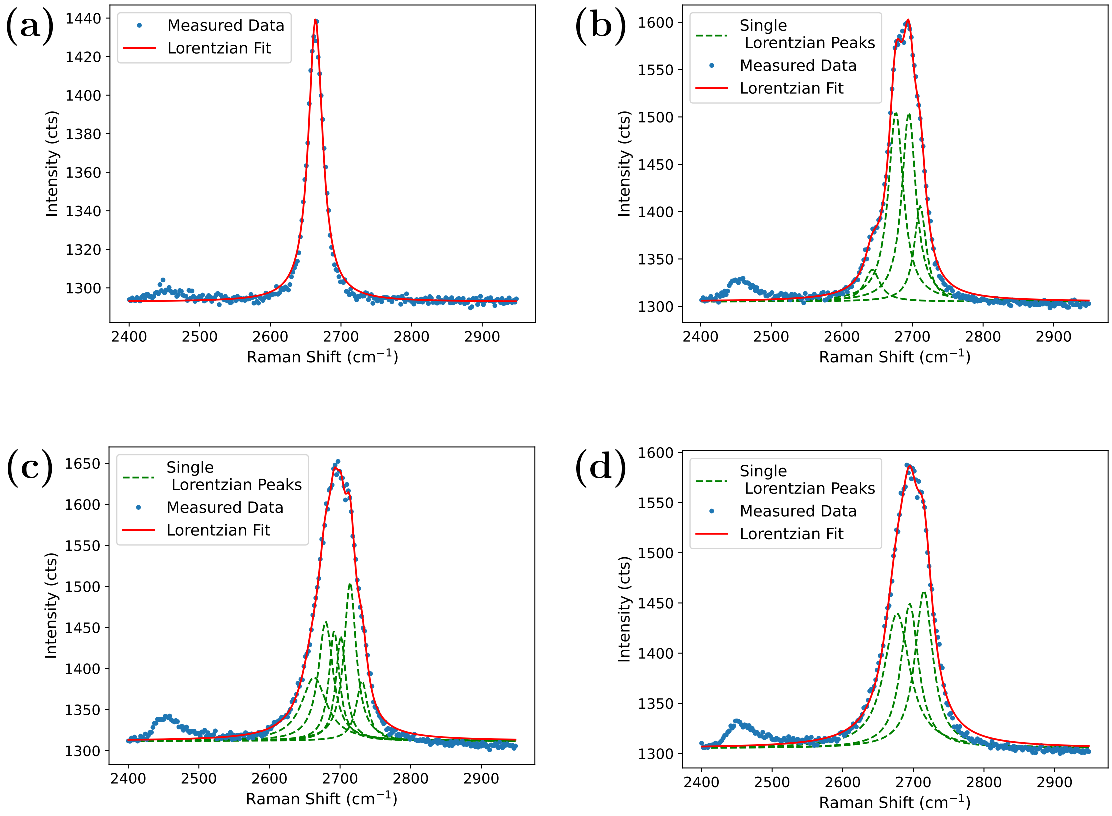


Figure 3.3: 2D-peak fitted with the minimum number of Lorentzian peaks required for (a) monolayer, (b) bilayer, (c) trilayer and (d) tetralayer

The data obtained from the WITec Suite SIX Software is exported and further analyzed using Python scripts, see appendix B. For the analysis, a single Lorentzian function is fitted to the G-peak and 2D-peak of the graphene Raman spectrum. Additionally, the 2D-peak is also fitted with the minimum number of Lorentzian peaks required for each layer number, as described in section 2.3, to accurately fit the Raman peak. Specifically, the measurements of the monolayers are fitted by a single Lorentzian, bilayers with four, trilayers with six, and all other layer numbers with three, as shown in Fig. 3.3.

The Lorentzian function ($f(x)$) used for peak fitting is given by:

$$f(x) = y_0 + \sum_{i=1}^{N} \frac{2A_i w_i^2}{4(x - x_{0,i})^2 + w_i^2}, \tag{3.1}$$

where y_0 is the base intensity, A is the amplitude of the i-th Lorentzian peak, w is the width, x_0 is the center of the peak and N being the number of required peaks for each fit.

To determine the ratio of the integrated intensities (I) of the G- and 2D-peak, the Lorentzians are integrated and calculated using the formula:

$$I = \sum_{i=1}^{N} \left(A_i w_i \arctan\left(\frac{2(x_1 - x_{0,i})}{w_i}\right) - A_i w_i \arctan\left(\frac{2(x_2 - x_{0,i})}{w_i}\right)\right), \tag{3.2}$$

where x_1 and x_2 are the boundaries of the integral. For the G-peak $x_1 = 1680\ \mathrm{cm}^{-1}$ and $x_2 = 1480\ \mathrm{cm}^{-1}$ are used, while for the 2D-peak, $x_1 = 3200\ \mathrm{cm}^{-1}$ and $x_2 = 2200\ \mathrm{cm}^{-1}$ are used.

3.4 Tapping Mode

To be able to compare the height measurements of the graphene flakes to the results of PeakForce^{TM} Microscopy, tapping mode images are taken with a Cypher L AFM from Oxford instruments Asylum Research, using the Ergo software in “Tapping” mode. For the imaging, the cantilevers Tap300Al-G from BudgetSensors are used, with a spring constant of 40 N/m and a tip radius of 10 nm provided by the manufacturers [63].

3.5 PeakForce^{TM} Microscopy

PeakForce^{TM} microscopy has been found to be able to accurately measure the height of monolayer graphene [25]. Here, graphene flakes up to hexalayer are measured to determine, if this method is reliable to characterize the number of layers and the height.

To perform the PeakForce^{TM} microscopy, a Bruker Dimension Icon Atomic Force Microscope is used in the ScanAssist in Air mode. As cantilevers the μmasch HQ:NSC18/Al BS are used, with a spring constant of 2.8 N/m and a tip radius of less than 8 nm given by the manufacturers [58].

Before starting the measurement, the deflection sensitivity and the spring constant are determined by using the thermal tune function of the NanoScope 6 controller. The deflection sensitivity and spring constant are calculated by the program using the “Defl Sens by Thermal” and “K by Thermal” function, as further elaborated in Chapter 2.5.4. More details of the measuring procedures can be found in Appendix A.

3.6 Analysis of the Tapping Mode and PeakForce™ Microscopy

To determine the measured height difference between the SiO_2/Si-substrate and the graphene flake, the acquired images are first analyzed using the software Gwyddion. Initially, the function “Align rows using various methods” is applied, with “Median differences” selected as the method, to level out the offset of each scanning line by shifting the line so that the median of height differences between the vertical lines is zero [64, 65]. Afterwards, the function “Level data by fitting a plane through three points” is used, where three points are placed on the free substrate, so that a plane is fitted through them and subtracted from the height measurements. This sets the height of the substrate to zero [64, 65]. The height distribution is then plotted using the “Calculate 1D statistical functions” function [64, 65].

The height distribution is extracted from Gwyddion and further analyzed in Python script C, where two Gaussian peaks are fitted to the distribution. By comparing the centers of the two Gaussian peaks, the height difference between the substrate and the graphene flake is determined (Fig. 3.4(b)).

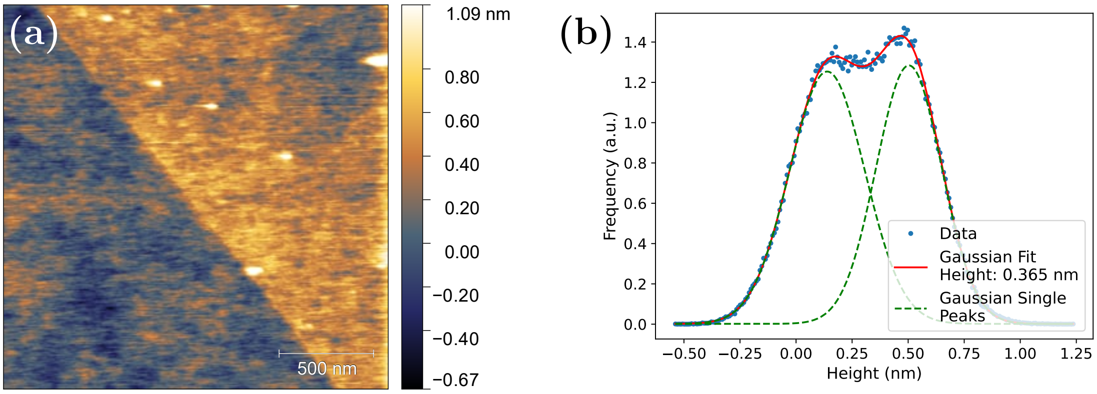


Figure 3.4: (a) Map of a part of a monolayer graphene flake taken with PeakForce™ with a setpoint of 15 nN (b) Graph of number of certain heights found in map, with two Gaussian peaks fit

3.7 Scanning Electron Microscopy

To assess whether PeakForce^{TM} microscopy damages the tips of the cantilevers during the measurements, images of the tips are taken, after using them with different maximum PeakForce^{TM} setpoints.

The scanning electron microscope (SEM) used to image the cantilever tips is a Zeiss Gemini DSM 982. All the images are taken with a voltage of 10 kV, with 3700x (Fig. 4.9) and 55000x (Fig. 4.8) magnification.

Chapter 4

Results and Discussions

The different methods, such as Raman spectroscopy, tapping mode and PeakForce$^{\text{TM}}$ microscopy, used to determine the number of graphene layers were analyzed, to assess their accuracy. For the following analysis, the number of layers identified by optical contrast was assumed to be correct.

4.1 Raman Spectroscopy

To identify the number of layers of graphene using Raman, multiple characterization methods were used, such as calculating the full width at half maximum and the center of the 2D-peak, as well as the ratio of the G-peak to the 2D-peak. For each method, the results were compared by using a single Lorentzian peak and the minimum number of Lorentzian peaks required to accurately fit the Raman peaks, to determine whether the more accurate fitting methods improve the ability to distinguish different layer numbers from each other or if the simpler method is sufficient. Additionally, the full width at half maximum values were compared with those reported in [11], see Fig. 4.1, to assess their consistency.

4.1.1 Full Width at Half Maximum

First, the full width at half maximum (FWHM) of the 2D-peak was calculated by fitting a single Lorentzian peak to it. Additionally, the FWHM was also calculated by fitting the minimum number of Lorentzian peaks required to accurately describe the 2D peak, as described in Chapter 3.3. The results of these two methods were compared with each other and the FWHM values for the different layer numbers reported in [11] (Fig. 4.2).

The measured monolayers had a mean FWHM of 26.6 cm^{-1}, with values ranging from 24.2 cm^{-1} to 28.3 cm^{-1}, all within the range of 27.5 ± 3.8 cm^{-1} given in [11]. In comparison, the bilayer FWHM values did not overlap with the range reported

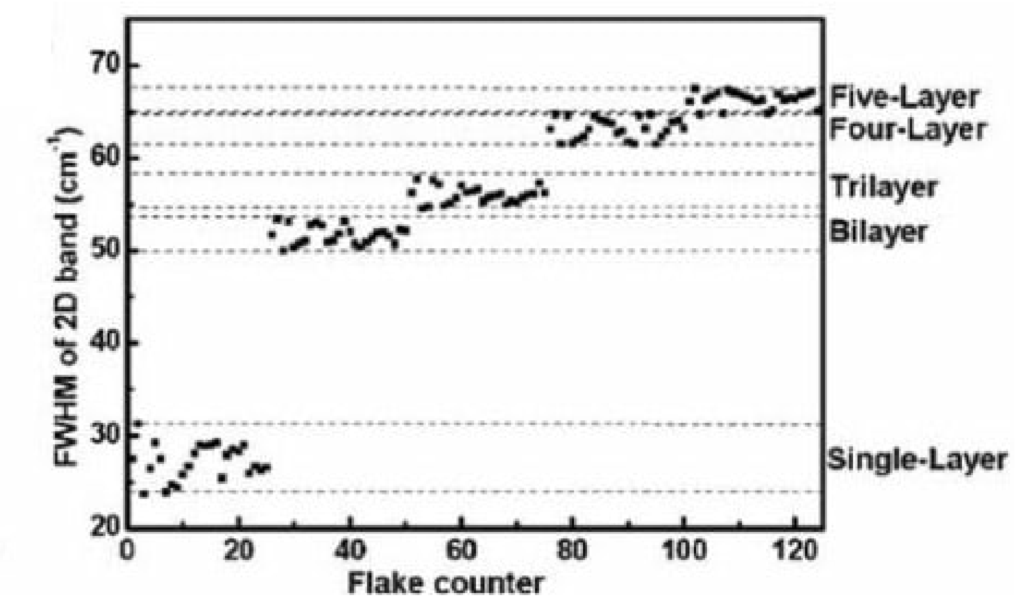


Figure 4.1: FWHM values in relation to the number of layers, as measured in [11], taken from Ref. [11]

in [11]. The bilayer FWHM, fitted with a single peak, ranged from 46.5 cm^{-1} to 48.4 cm^{-1} with a mean of 47.3 cm^{-1}. This was significantly lower than the reported value 51.7 ± 1.7 cm^{-1}. The FWHM values, fitted with four peaks, for the bilayer ranged from 47.0 cm^{-1} to 54.1 cm^{-1}, with a mean of 51.7 cm^{-1}, agreeing well with the mean value reported in [11].

For the trilayers fitted with a single peak, the mean FWHM was 55.3 cm^{-1}, with a range of 53.7 cm^{-1} to 56.8 cm^{-1}, being half above and half within the range of 56.2 ± 1.6 cm^{-1} from [11]. The FWHM values for the trilayer fitted with six peaks, however, consistently lay above the reported range of [11], with values ranging from 61.1 cm^{-1} to 65.1 cm^{-1} and a mean value of 63.1 cm^{-1}.

All the FWHM values for the tetralayers, fitted with a single Lorentzian peak, fell below the reported range of 63.1 ± 1.6 cm^{-1} in [11]. They ranged from 55.7 cm^{-1} to 60.1 cm^{-1}, even though the value of 60.1 cm^{-1} lay quite far from the other values, which had a much smaller range from 55.7 cm^{-1} to 56.9 cm^{-1} with a mean of 57.4 cm^{-1} including the value of 60.1 cm^{-1}. While FWHM values, fitted with three Lorentzian peaks, lay above the range with the exception of one value which lay within the range. The mean FWHM for the tetralayers, fitted with three Lorentzian peaks, was 64.8 cm^{-1} with a range of 62.1 cm^{-1} to 69.1 cm^{-1}. Similar to FWHM, fitted with a single Lorentzian, the range was much smaller, if the value of 69.1 cm^{-1} was disregarded, 62.1 cm^{-1} to 64.1 cm^{-1}.

For pentalayer samples, the mean of the FWHM, fitted by a single Lorentzian peak, was 57.9 cm^{-1}, with the measured values 57.2 cm^{-1} and 58.6 cm^{-1}, falling below the reported range. The FWHM values of the pentalayer, fitted with three

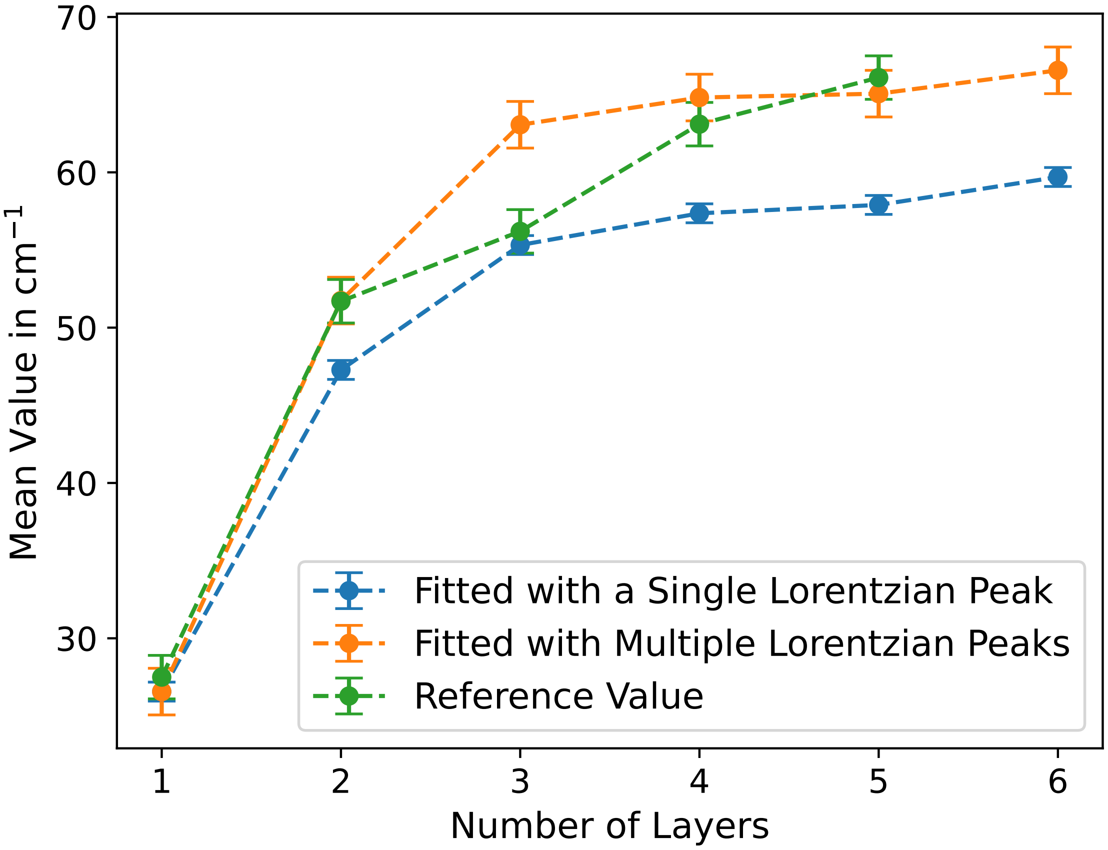


Figure 4.2: The mean values from both fitting methods plotted against the number of layers determined by optical contrast with the error bars representing the standard deviations of values measured, as well as the values given by [11] and the ranges given in [11] as error bars

peaks, 64.1 cm^{-1} and 66.1 cm^{-1}, fell within the reported range of 66.1 ± 1.4 cm^{-1} from [12], with the mean value being 65.1 cm^{-1}. For the hexalayer FWHM the literature does not mention any values. The FWHM values, fitted with a single peak, were 59.1 cm^{-1} and 60.3 cm^{-1} with the mean of 59.7 cm^{-1} and the FWHM values, fitted with three, were 65.1 cm^{-1} and 68.1 cm^{-1} with the mean of 66.6 cm^{-1}. An overview of the values is displayed in Table 4.1.

The literature value had the most overlap with the evaluation method using the minimum number of Lorentzian peaks necessary to fit the 2D-peak. Unlike in [11], the FWHM intervals for samples with more than two layers began to overlap, which made it difficult to characterize them with this method. Despite this, mono- and bilayers were well distinguishable, when comparing the FWHM using both fitting methods. Interestingly, when examinating the standard deviations the FWHM fitted with a single Lorentzian peak appeared to be more reliable than the one fitted with multiple Lorentzian peaks, as the standard deviations were smaller.

Number of Layers	**Value**	**FWHM single in cm^{-1}**	**FWHM multi in cm^{-1}**	**Literature value in cm^{-1}**
1	Mean	26.6	–	27.5
	Std Dev	1.3	–	3.8
	Min	24.2	–	
	Max	28.3	–	
2	Mean	47.3	51.7	51.7
	Std Dev	0.5	2.2	1.7
	Min	46.5	47.0	
	Max	48.4	54.1	
3	Mean	55.3	63.1	56.2
	variance	1.5	1.6	1.6
	Min	53.7	61.1	
	Max	56.8	65.1	
4	Mean	57.4	64.8	63.1
	Std Dev	1.7	2.6	1.6
	Min	55.7	62.1	
	Max	60.1	69.1	
5	Mean	57.9	65.1	66.1
	variance	0.8	1.0	1.4
	Min	57.2	64.1	
	Max	58.6	66.1	
6	Mean	59.7	66.6	n/a
	Std Dev	0.7	1.5	n/a
	Min	59.1	65.1	
	Max	60.3	68.1	

Table 4.1: Mean value (Mean), standard deviation (Std Dev), and minimum (Min) and maximum (Max) measured values for the FWHM fitted with a single Lorentzian peak (FWHM single) and the minimum number of required Lorentzian peaks to fit the 2D peak accurately (FWHM multi), as well as the mean value and the standard deviation reported in [11] (Literature value)

4.1.2 2D-Peak Position

As it was previously shown in [40], the values of the 2D-peak position increase with the number of layers, as depicted in Fig. 4.3. Therefore in this section, it was explored, whether the 2D-peak position could be used to identify the number of layers. This additional analysis was performed, since the analysis of the FWHM was only able to reliably identify the number of layers of monolayer and bilayer graphene. The position of the 2D-peak was compared to the number of layers determined by optical contrast. The center of the peak was extracted from the same fits used to calculate the FWHMs. As with the FWHM, the data of the fits performed with a single Lorentzian peak and those using the number of Lorentzian peaks required for each specific layer, as described in Chapter 3.3, were compared.

For monolayers, the centers of the 2D-peaks ranged from 2658.5 cm^{-1} to 2670.6 cm^{-1},

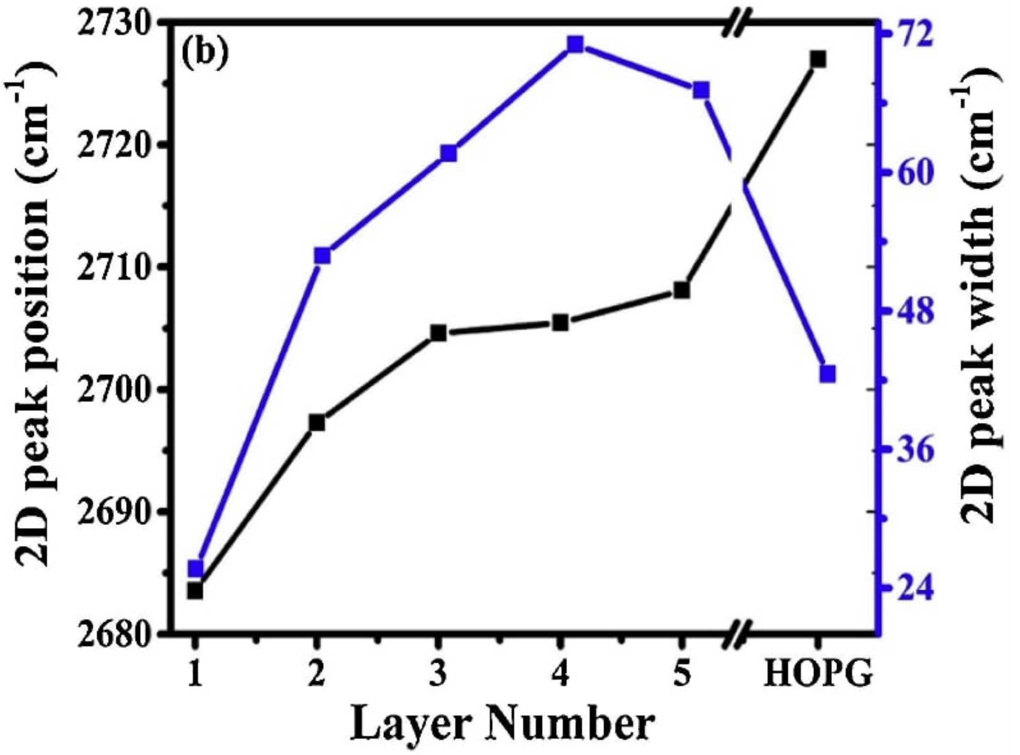


Figure 4.3: The black line represents the 2D-peak position depending on the number of layers measured using a 514.5 nm wavelength laser reported in [40]. Taken from Ref. [40]

with a mean value of 2664.6 cm^{-1}. The monolayers could be distinguished well from the other layer numbers, as their range did not overlap with any other layer, regardless of the fitting method. They were separated by 11.6 cm^{-1} from the bilayers fitted with a single Lorentzian peak and by 13.0 cm^{-1} from the bilayers fitted with four Lorentzian peaks.

For the bilayers fitted with a single Lorentzian peak, the 2D-peak centers ranged from 2682.2 cm^{-1} to 2691.0 cm^{-1}, with a mean value of 2686.7 cm^{-1}. This range overlapped with that of the trilayers, which were also fitted using a single Lorentzian peak. When the bilayers were fitted with four Lorentzian peaks, the centers ranged from 2683.0 cm^{-1} to 2691.5 cm^{-1} and had a mean value of 2687.3 cm^{-1}. In contrast to the single Lorentzian peak fit, the resulting centers from the four Lorentzians did not overlap with any other layer number, even though the range of the trilayers, fitted by six Lorentzian peaks, and bilayers were only separated by 0.6 cm^{-1}.

For the trilayers fitted by a single Lorentzian peak, the 2D-peak centers ranged from 2690.2 cm^{-1} to 2696.4 cm^{-1}, with a mean value of 2692.6 cm^{-1}. This range overlapped with that of the centers of the bi-, tetra- and pentalayers, which were fitted with a single Lorentzian peak as well, showing that this method is only reliable to distinguish monolayers from the other layers. In contrast, the range of the trilayers fitted by six Lorentzian peaks narrowed to 2692.0 cm^{-1} to 2696.0 cm^{-1}, with a mean center of 2694.0 cm^{-1}. While this fitting method removed the overlap with the bilayers, the range still overlapped with the tetra- and pentalayers, fitted with three Lorentzian peaks and additionally started to overlap with the range of the hexalayers. Showing that this fitting method also cannot differentiate between layers that are higher than 2.

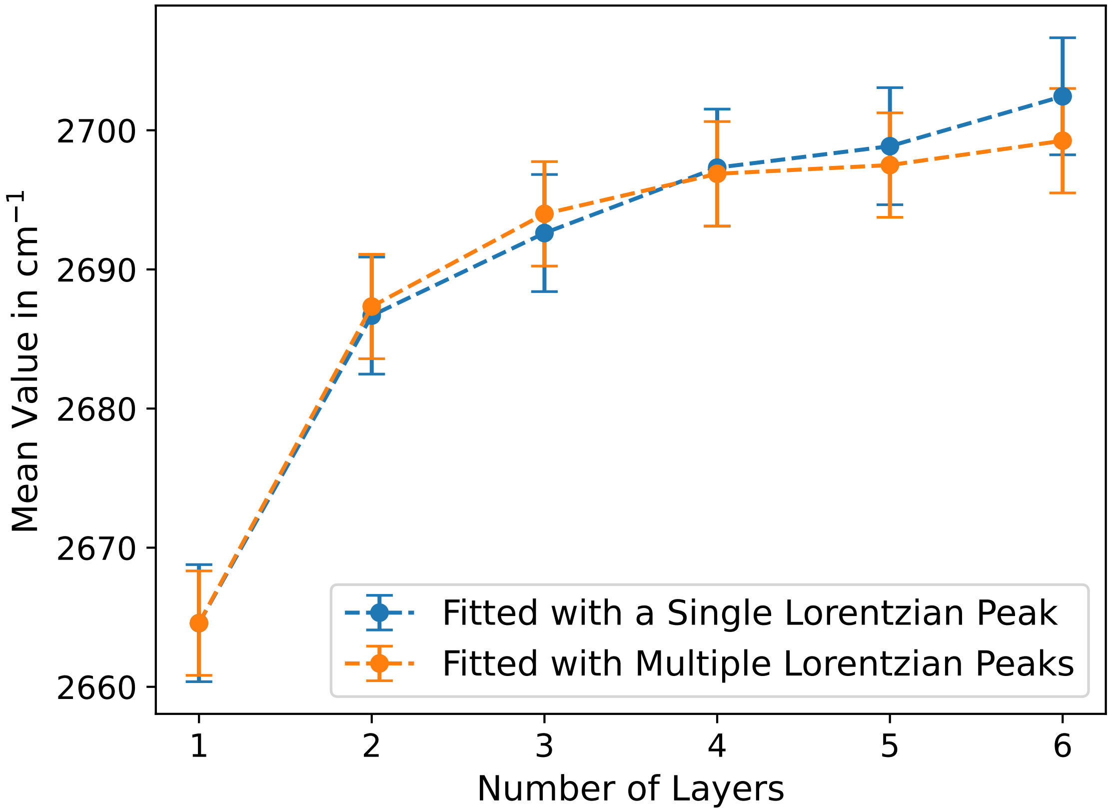


Figure 4.4: 2D-peak center mean values and ranges determined by fitting the peak with a single Lorentzian peak (2D-peak center single) and by the number of Lorentzian peaks required for each specific layer (2D-peak center multiple), as described in Chapter 3.3, plotted against the number of layers found with optical contrast. The error bars show the standard deviations of the measured values.

For the tetralayers fitted by a single Lorentzian peak, the 2D-peak centers ranged from 2696.3 cm^{-1} to 2697.8 cm^{-1}, with a mean of 2697.3 cm^{-1}. When fitted with three Lorentzian peaks, the tetralayers ranged from 2695.5 cm^{-1} to 2698.0 cm^{-1}, with a mean of 2697.5 cm^{-1}. For the two pentalayer flakes, the center of the peaks lay at 2694.4 cm^{-1} and 2703.3 cm^{-1}, when fitted with a single Lorentzian peak, and had a mean of 2698.9 cm^{-1}. When fitted with three Lorentzian peaks, the center of the peaks were at 2693.5 cm^{-1} and 2701.5 cm^{-1}, with a mean of 2697.5 cm^{-1}. The two hexalayers fitted by a single Lorentzian peak, had peaks at 2698.2 cm^{-1} and 2706.7 cm^{-1}, with a mean of 2702.4 cm^{-1}. While the values lay at 2695.5 cm^{-1} to 2703.0 cm^{-1}, and had a mean of 2699.2 cm^{-1}, when fitted with three Lorentzian

Number of Layers	**Value**	**2D center single in cm^{-1}**	**2D center multi in cm^{-1}**
1	Mean	2664.6	-
	Std Dev	4.6	-
	Min	2658.5	-
	Max	2670.6	-
2	Mean	2686.7	2687.3
	Std Dev	2.8	2.7
	Min	2682.2	2683.0
	Max	2691.0	2691.5
3	Mean	2692.6	2694.0
	Std Dev	2.4	1.6
	Min	2690.2	2692.0
	Max	2696.4	2696.0
4	Mean	2697.3	2697.5
	Std Dev	0.7	1.0
	Min	2696.3	2695.5
	Max	2697.8	2698.0
5	Mean	2698.9	2697.5
	Std Dev	4.5	4.0
	Min	2694.4	2693.5
	Max	2703.3	2701.5
6	Mean	2702.4	2699.2
	Std Dev	4.3	3.8
	Min	2698.2	2695.5
	Max	2706.7	2703.0

Table 4.2: 2D center mean value (Mean), standard deviation (Std Dev), and minimum (Min) and maximum (Max) measured values fitted with a single Lorentzian peak (2D center single) and with the minimum number of required Lorentzian peaks to fit the 2D peak accurately (2D center multi)

peaks. The tetralayers ranges for both methods overlapped with the trilayers and were completely within the pentalayer range. The range fitted with the 3 Lorentzian peaks also overlapped with the hexalayers, whereas the range of the hexalayers only overlapped with the pentalayers for the fits done with a single Lorentzian peak. See Table 4.2 for an overview of the measured values.

Although both fitting methods can differentiate monolayers from other layers, fitting with the Lorentzian peaks required for each specific layer makes it possible to separate bilayers from higher layers as well. Both methods are not able to distinguish between the higher layers.

4.1.3 Ratio between the Integrated Intensities of the G- and 2D-Peaks

Similar to Section 4.1.2, in [47] the ratio of the integrated intensity of the G-peak to that of the 2D-peak also depends on the number of layers, as can be seen in Fig. 4.5. Therefore, the ratio of the integrated intensity of the G-peak to that of the 2D-peak was analyzed in relation to the number of graphene layers, for both fitting methods of the 2D-peak. For the monolayers, the ratio of the integrated intensities of the G-peak and 2D-peak ranged from 0.32 to 0.54 and had a mean value of 0.42.

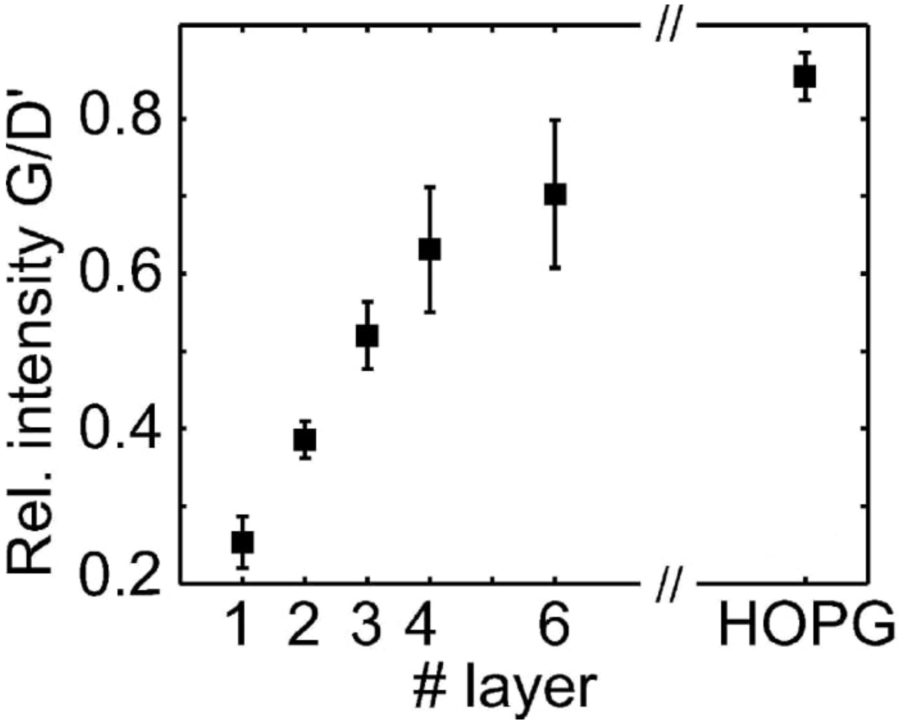


Figure 4.5: Integrated intensity ratio of the G-peak and 2D-peak found in [47]. These values are not comparable to the ones measured in this thesis, since the integration intervals used in this thesis differ from the ones used in [47]. Taken from Ref. [47]

One sample of the bilayers had a ratio that fell within the range of monolayers when using a single Lorentzian peak, with a value of 0.42, as well as when using four Lorentzian peaks, with a value of 0.48. The other bilayers lay within a range of 0.67 to 0.81, with a mean of 0.72, for the single Lorentzian peak and within 0.74 to 0.89, with a mean of 0.77, for the four Lorentzian peaks and did not overlap with the ranges of the other layers.

The trilayer values were well grouped but could not be distinguished from the tetralayers and pentalayers, as the trilayers lay fully within the measured range of the tetralayers and pentalayers. The range of the trilayers was from 0.94 to 0.98, with a mean of 0.96, for the single Lorentzian peak and from 1.07 to 1.11, with a mean of 1.09, for the six Lorentzian peaks.

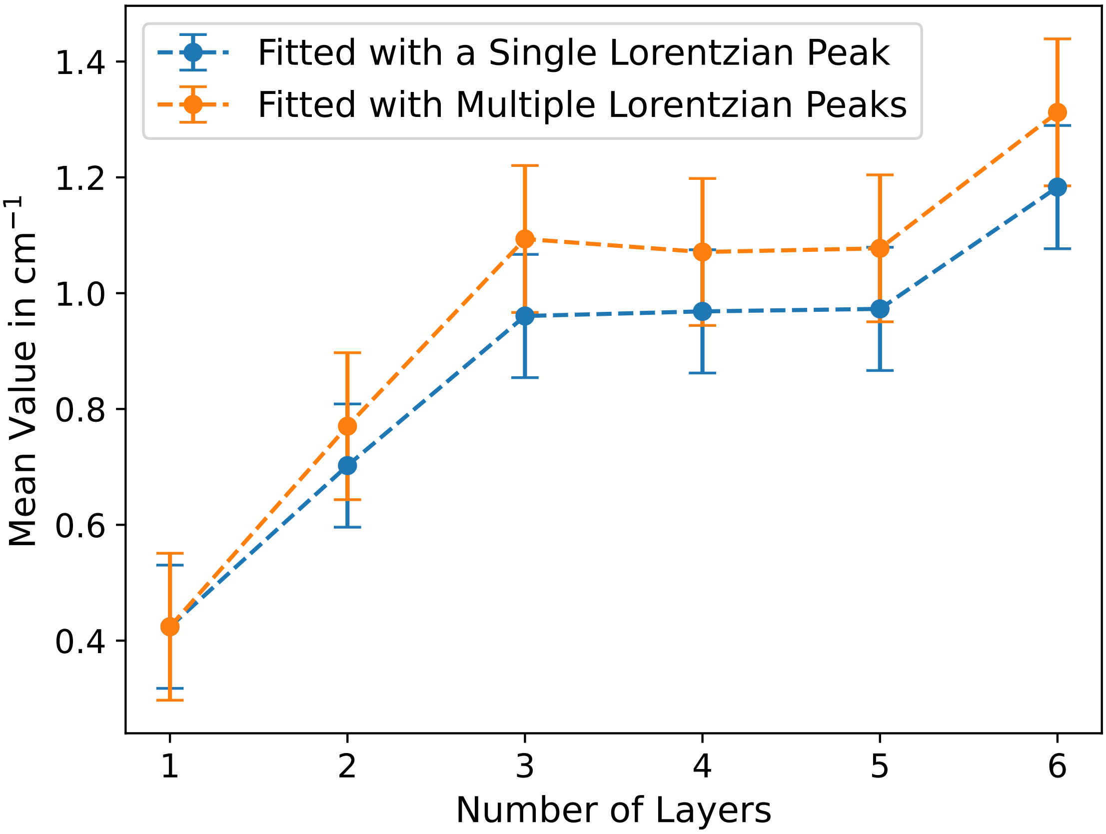


Figure 4.6: Mean values of the ratio of the integrated intensity of the G-peak and the 2D-peak plotted against the number of layers identified by optical contrast. The 2D-peak was fitted with a single Lorentzian peak as well as with the number of Lorentzian peaks required, as specified in Chapter 3.3. The error bars are the standard deviations of the measured values.

The tetralayer values ranged from 0.81 to 1.10 for the single Lorentzian peak, with a mean of 0.97 and from 0.91 to 1.21 for the three Lorentzian peaks, with a mean of 1.07. The pentalayer flakes, the two measured values were 0.89 and 1.06 with a mean of 0.97 for the fit using the single Lorentzian and 0.98 and 1.17 with a mean of 1.08 for the fit using three. The mean ratios of the tetra- and pentalayers were very close to each other for both fitting methods, for the fitting method using only one Lorentzian peak, the ratios were also close to the ratio of the trilayers. For the fitting method using three Lorentzian peaks, the mean ratios of the tetra- and pentalayers were below the ratio of the trilayers.

The two hexalayers had a ratio of 1.08 and 1.29, with a mean of 1.18, when fitted with a single Lorentzian peak and a ratio of 1.19 and 1.44, with a mean of 1.31, when fitted with three Lorentzians. The ratios measured for the hexalayers lay very far apart, and overlapped with the ratio of the tetralayers. An overview of the values can be found in Table 4.3.

By analyzing the ratio between G- and 2D-peak, it is possible to distinguish mono- and bilayers from other number of layers. However, to differentiate between three or more layers is not possible and distinguishing between mono- and bilayers also comes with uncertainty. The accuracy for determining the number of layers did not improve for one fitting method over the other.

Number of Layers	Value	G/2D ratio single	G/2D ratio multi
1	Mean	0.42	-
	Std Dev	0.08	-
	Min	0.32	-
	Max	0.54	-
2	Mean	0.72	0.77
	Std Dev	0.09	0.10
	Min	0.43	0.48
	Max	0.81	0.89
3	Mean	0.96	1.09
	Std Dev	0.19	0.02
	Min	0.94	1.07
	Max	0.98	1.11
4	Mean	0.97	1.07
	Std Dev	0.45	0.12
	Min	0.81	0.91
	Max	1.10	1.21
5	Mean	0.97	1.08
	Std Dev	0.45	0.10
	Min	0.89	0.98
	Max	1.06	1.17
6	Mean	1.18	1.31
	Std Dev	0.45	0.13
	Min	1.08	1.19
	Max	1.29	1.44

Table 4.3: G peak to 2D peak ratio mean value (Mean), standard deviation (Std Dev), and minimum (Min) and maximum (Max) measured values, where the 2D is fitted with a single Lorentzian peak (G/2D ratio single) and with the minimum number of required Lorentzian peaks to fit the 2D peak accurately (G/2D ratio multi), while the G peak is always fitted with a single Lorentzian peak

4.1.4 Comparison of Raman Analysis Methods

Most of the analysis methods could differentiate mono- and bilayers from each other and the other layers. The most reliable methods were examining the FWHM and the center of the 2D-peak, although the latter was only among the most reliable when the 2D-peak was fitted with the necessary number of Lorentzian peaks. The flakes that were identified as having a different number of layers than determined by optical contrast with one method, fell firmly within the range matching the optical contrast for the other methods. This confirmed that optical contrast accurately determined the number of layers for monolayers and bilayers.

However, the higher layers could not be determined using Raman spectroscopy, regardless of the analysis method, since there was no recognizable pattern. It was observed that values that had a low value with one measurement method, for example, had a high value with another measurement method compared to samples with the same number of layers and vice versa.

The fitting method, using the number of Lorentzian peaks that were minimally required depending on the number of layers, only slightly improved the differentiability of the layers. The Raman analysis method, that was the most effective is the FWHM, since even the simpler fitting method using only one Lorentzian peak, was able to reliably distinguish between monolayers and bilayers.

4.2 Tapping Mode

In total only twelve graphene flakes were measured using the AFM in tapping mode, since it is already well known that this method tends to overestimate the height of the graphene flakes and these measurements were only conducted to be compared to PeakForce$^{\mathrm{TM}}$ microscopy. Of these twelve graphene flakes, four monolayers, four bilayers, two trilayers and two tetralayer were measured.

The monolayers height measurements had a large range from 0.260 nm to 1.025 nm and a mean height of 0.652 nm, which was almost double the theoretical height of a graphene monolayer of 0.335 nm [25].

The height measurements of the bilayers also had a large range from 1.094 nm to 1.858 nm and a mean height of 1.399 nm. All of the bilayer height measurements significantly overestimated the height. The measured height ranges of the bilayers and the monolayers did not overlap, making it possible to differentiate between them, although given the large uncertainty of the measurements and small separation between them, this distinction could not be considered reliable.

The measured trilayer heights ranged from 1.636 nm to 2.020 nm, with a mean of 1.828 nm, while the tetralayer heights ranged from 0.877 nm to 1.768 nm, with a mean of 1.323 nm (Table 4.4). The height measurement ranges for the bilayer, trilayer, and tetralayer overlapped, making it impossible to distinguish between them.

Additionally, for most of the layer numbers, the tapping mode measurements exhibited a standard deviation, that was about half of the monolayer's measured height, further highlighting their high uncertainty and rendering them unsuitable for determining the number of layers. Further, the measured heights often overestimated the height of the flake, when compared to the expected theoretical values.

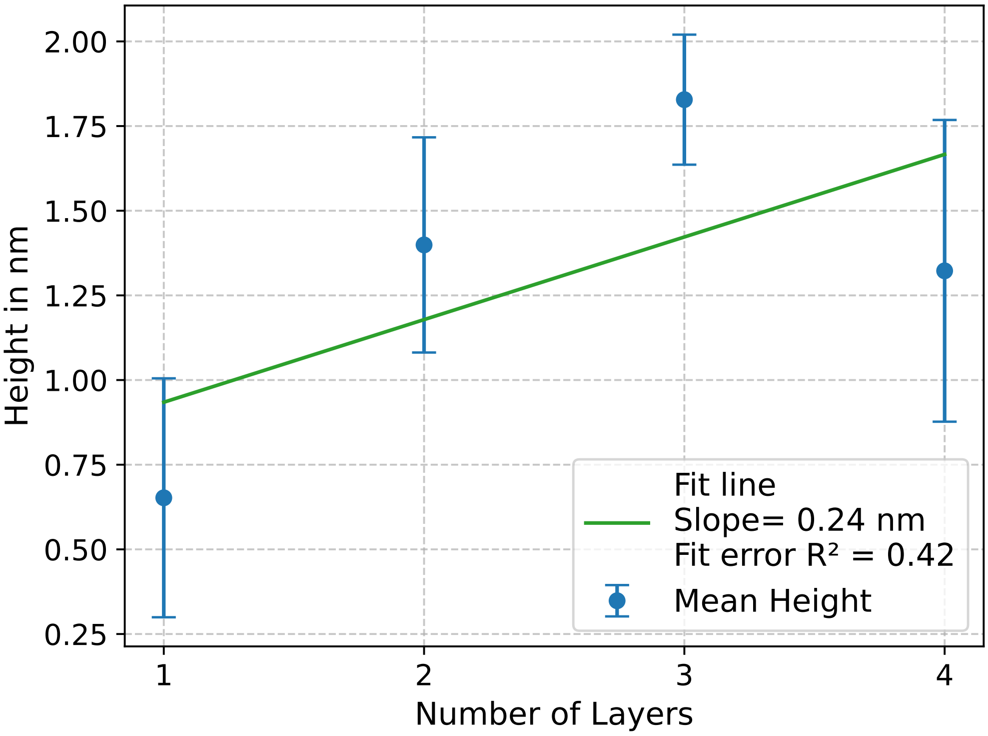


Figure 4.7: Mean height tapping mode measurements, with their standard deviation represented as error bars, plotted against the number of layers determined by optical contrast, and a line fitted through them

Layers	**Mean in nm**	**Std Dev in nm**	**Min in nm**	**Max in nm**
1	0.65	0.35	0.26	1.025
2	1.40	0.32	1.094	1.858
3	1.83	0.19	1.636	2.020
4	1.32	0.45	0.877	1.768

Table 4.4: Mean value (Mean), standard deviation (Std Dev), and minimum (Min) and maximum (Max) measured values obtained using tapping mode

4.3 PeakForce^{TM} Microscopy

This section describes the challenges encountered with PeakForce^{TM} microscopy and their solutions while developing a procedure to measure the height of graphene flakes for layer number identification. Additionally, the analysis of the measured height in relation to the number of layers is presented and the reliability of this method for determining the number of layers is evaluated.

4.3.1 Difficulties with Thermal Tune

While using the thermal tune function, it was often observed that after acquiring the data for the first time, it was not possible to calculate the spring constant with the “K by Thermal” function. This issue could be resolved by calculating the deflection sensitivity first with the “Defl Sens by Thermal” function and then acquiring the data again. If acquiring the data twice is not desired, the issue can also be addressed by manually inputting a deflection sensitivity value close to the one of the cantilever.

Additionally, it was also noticed that even when the program could calculate the spring constant, if the deflection sensitivity before acquiring the data deviated strongly to the actual deflection sensitivity, the spring constant calculated by the program would vary a lot from the value given by the manufacturers. The calculated spring constants ranged from 2.1 to 4.1 N/m, while the value provided by the manufacturers was 2.8 N/m.

When acquiring the data a second time and allowing the program to calculate the data again, the resulting spring constants were closer to the values given by the manufacturers, between 2.7 N/m and 2.9 N/m. As repeating this process a third time did not change the spring constant value significantly, and the deflection sensitivity also only changed minimally after calculating it a second time, this suggests that the deflection sensitivity is sufficiently accurate after the first calculation. Therefore, performing more than two iterations did not improve the accuracy of the calculated spring constant.

4.3.2 Initial Measurements with Large PeakForce$^{\text{TM}}$ Setpoint Range

A graphene monolayer was first measured with five different PeakForce$^{\text{TM}}$ Setpoints, 10 nN, 25 nN, 50 nN, 75 nN and 100 nN, incrementing from 10 nN up to 100 nN and then decreasing back to 10 nN. During this measurement, the spring constant was still determined, while only measuring once, meaning that the spring constant might have strongly deviated during these measurements. This was done, in order to explore the range of viable forces applied to the cantilever tips before fracturing them by varying the setpoint and subsequently imaging them under SEM. It was observed that the height values of the monolayer dropped below its theoretical height when measured at 75 nN for the first time. This suggested that the tip might have broken during this measurement, due to a too high PeakForce$^{\text{TM}}$ Setpoint.

To confirm these observations, new measurement series were carried out with two different tips, with the maximum PeakForce$^{\text{TM}}$ Setpoint limited to 75 nN for the first tip and to 50 nN for the second. After measuring with a force of 75 nN, the image quality of the subsequent 50 nN measurement got significantly worse than the initial 50 nN measurement, indicating tip degradation. In contrast, no such degradation occurred when measuring with a maximum force of 50 nN.

To confirm the assumption that the tip broke during measurements with forces of 75 nN or higher, the tips were imaged using SEM. The SEM images confirmed visible damages to the tips used with the maximum PeakForce^{TM} Setpoints of 100 nN (Fig. 4.8(d), 4.9(d)) and 75 nN (Fig. 4.8(c), 4.9(c)), while the tip that was used with a maximum force of 50 nN (Fig. 4.8(a), 4.9(a)) showed no visible damage.

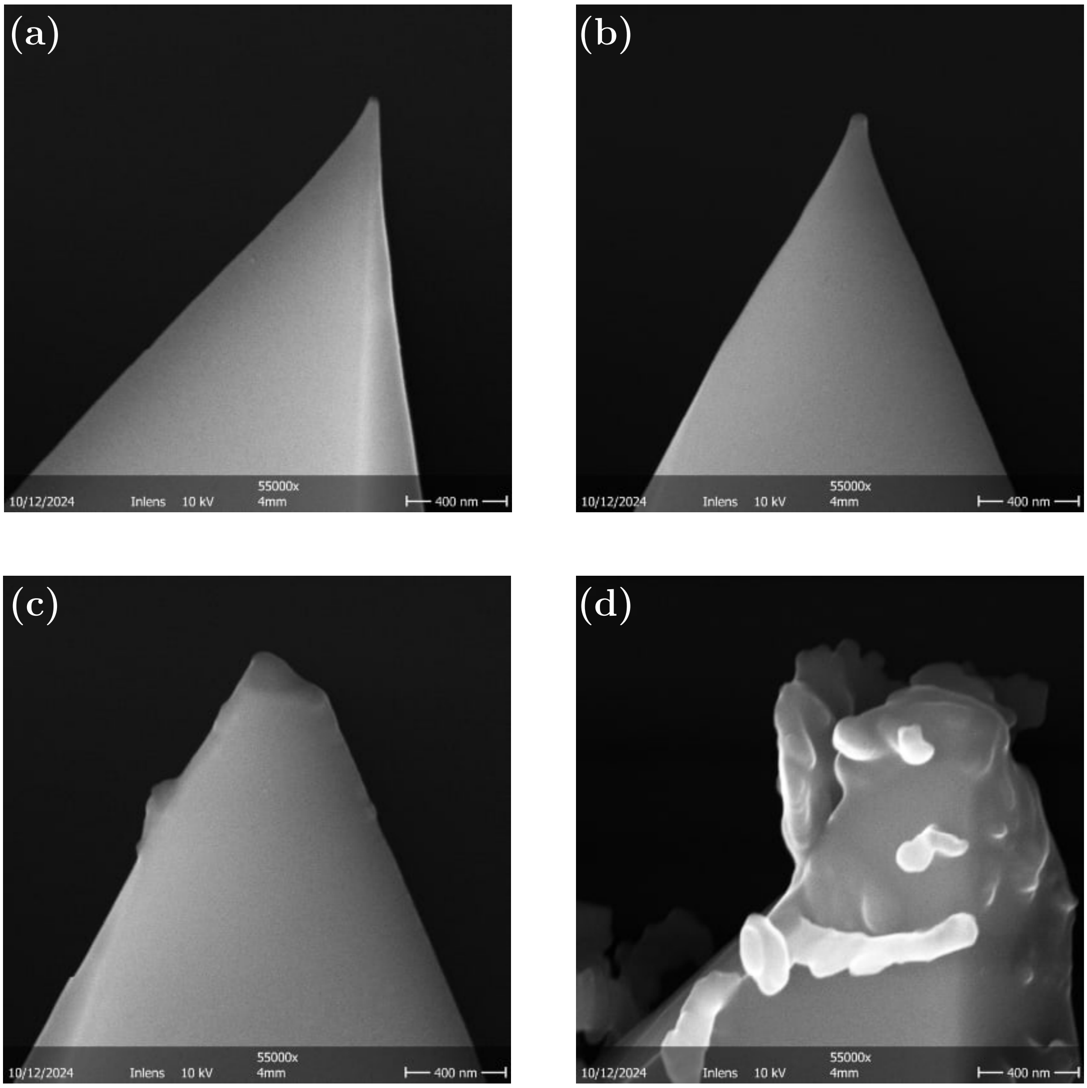


Figure 4.8: SEM pictures of the tips at a magnification of 55000x, that were used with the maximum PeakForce^{TM} Setpoints of (a) 50 nN, (b) 62.5 nN, (c) 75 nN and (d) 100 nN.

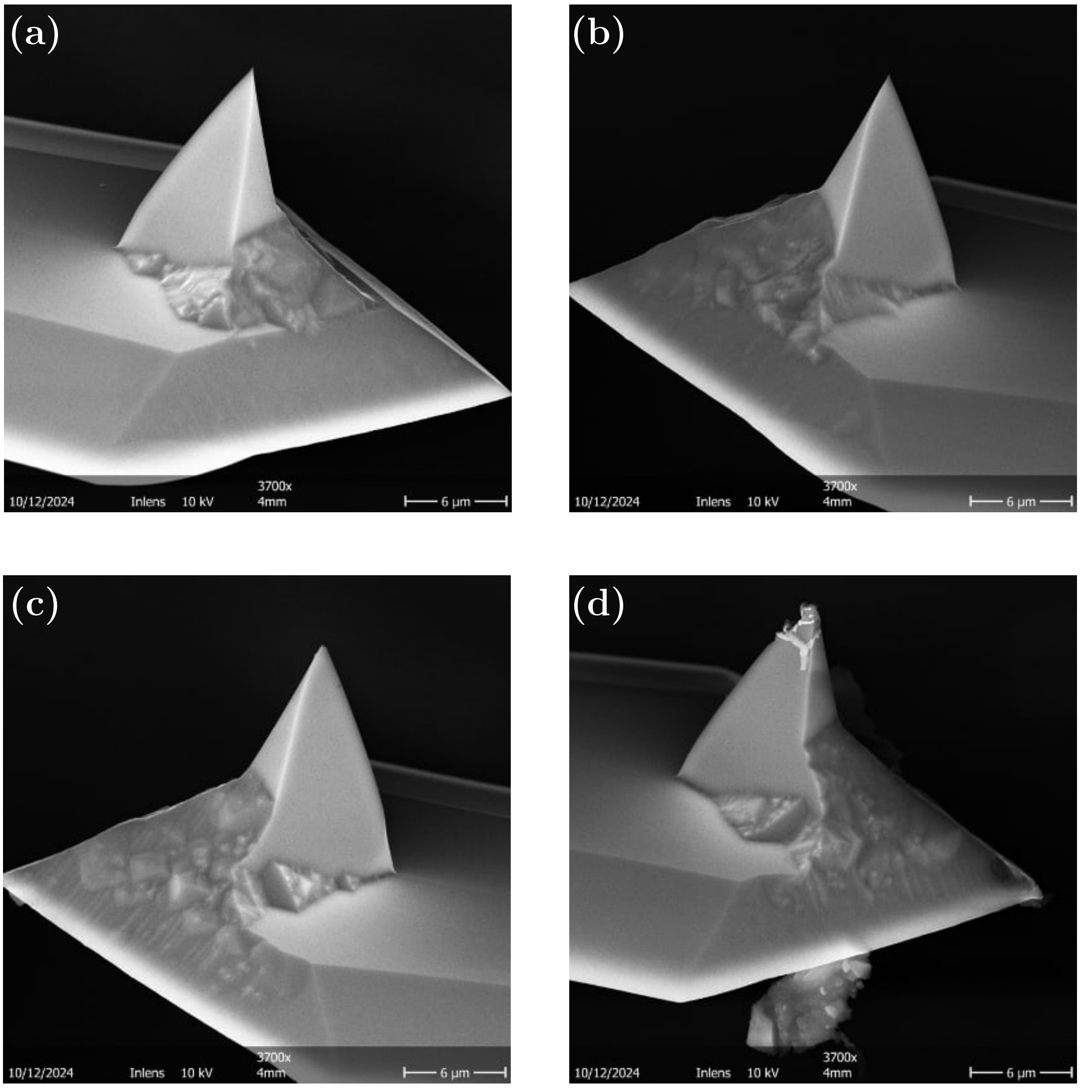


Figure 4.9: SEM pictures of the tips at a magnification of 3700x, that were used with the maximum PeakForce^{TM} Setpoints of (a) 50 nN, (b) 62.5 nN, (c) 75 nN and (d) 100 nN.

4.3.3 PeakForce^{TM} Measurements between the Setpoints 2,5 and 25 nN

A series of measurements was taken on the same flake at the same location with the PeakForce^{TM} setpoint increased in steps of 5 nN, 7.5 nN, 10 nN, 12.5 nN, 15 nN, 20 nN and 25 nN, sometimes including 2.5 nN. The force was always increased just one step lower than the previous measurement and then measured by going back down the steps. There was no noticeable pattern of measurements varying in any particular direction as the force was reduced.

Since the spring constant in this measurement was not accurately determined, it was analyzed if the measured height would be estimated lower or higher depending on the overestimation or underestimation of the spring constant by the program.

The correct values of the spring constant were assumed to be between 2.7 nN and 2.9 nN, since these were the values determined, when measuring with the more accurate method as described in Chapter 4.3.1. However, there did not seem to be a correlation between the measured spring constant and the measured height, as seen in Fig. 4.10.

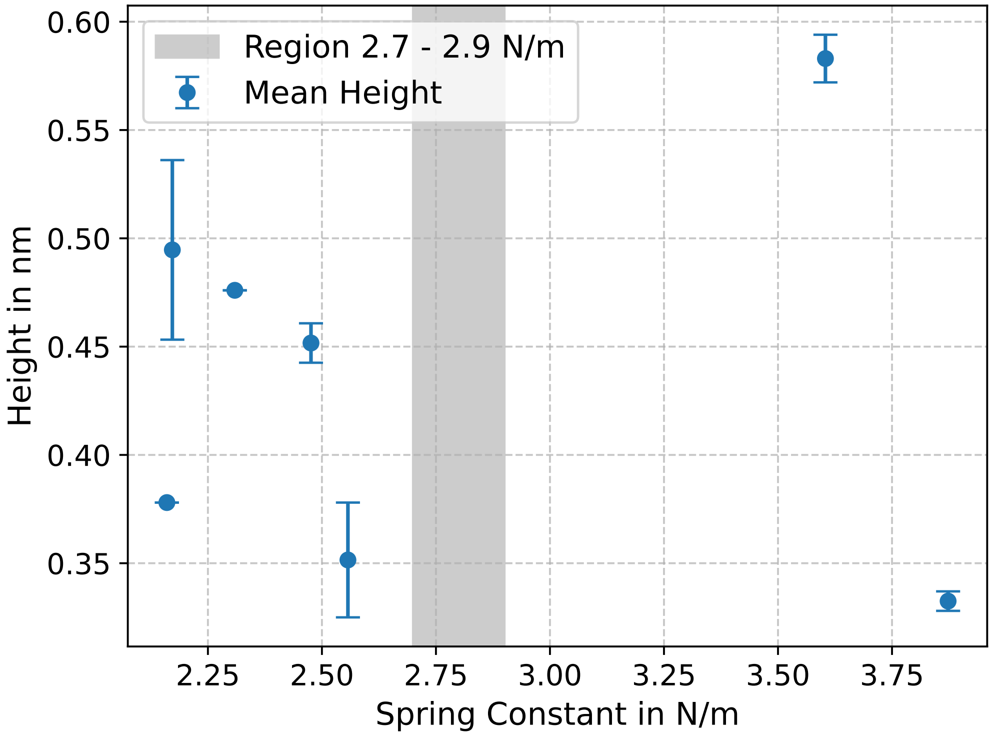


Figure 4.10: Height measurements of the same spot on a flake performed at a PeakForce^{TM} setpoint of 5 nN depending on the spring constant. The gray region displays the range of the spring constant values of the cantilevers, when measured using the more accurate method as described in Chapter 4.3.1. The error bars display the standard deviations for the measurements performed by the same cantilever.

4.3.4 Determination of the Optimal PeakForce^{TM} Setpoint

To determine the best PeakForce^{TM} setpoint for obtaining consistent height values, that were not significantly higher than the theoretical height of graphene, three monolayers and six bilayers were measured with the PeakForce^{TM} setpoints 2.5 nN, 5 nN, 7.5 nN, 10 nN, 12.5 nN, 15 nN, 20 nN and 25 nN. Occasionally increasing up to 37.5 nN, 50 nN and 62.5 nN, to check if the tip would be damaged and if high PeakForce^{TM} setpoints would compress the space between the layers, decreasing the height of the bilayers below their theoretical value. The optimal PeakForce^{TM} setpoint was determined by calculating the mean height and variance for each PeakForce^{TM} setpoint.

For the measured monolayers, the average measured height decreased, as the applied force increased from 2.5 nN to 5 nN, followed by a sharp rise to a peak at 7.5 nN, before it decreased again until 15 nN, reaching a minimum average height value of 0.431 nm. Beyond that, the height started to rise again (Fig. 4.11 (a)).

For the bilayer, it began similarly, with the average measured height decreasing from 2.5 nN to 5 nN, followed by an increase to a local peak at 7.5 nN, after which the average measured height decreased again. However, unlike the monolayer, the bilayer reached its minimum value of 0.660 nm only at 25 nN, after which the height started to increase again (Fig. 4.11 (b)).

The variance of the monolayer was generally quite low, except for 7.5 nN, where it was noticeably higher than the other variances, reaching 0.036 nm, while the next highest variance was only at 0.018 nm. This can be attributed to one of the measurements for 7.5 nN that was noticeably higher than the rest of the measurements, with a value of 0.99 nm, which might have been caused by a measurement error. The variance for the PeakForce$^{\mathrm{TM}}$ setpoints of 2.5 nN, 10 nN, 12.5 nN, 20 nN and 25 nN all lay around the value of 0.013 nm. The setpoints of 5 nm and 15 nm were the minima of this curve, with a variance of 0.001 nm and 0.003 nm, respectively.

For the bilayer, the variance remained relatively consistent across the PeakForce$^{\mathrm{TM}}$ setpoints from 2.5 nN to 7.5 nN and from 25 nN to 37.5 nN, with them averaging around 0.043 nm. In contrast, the variance decreased significantly at setpoints from 10 nN to 20 nN, where it was approximately half the value observed in the other ranges. The lowest variance was recorded at 10 nN, with a value of 0.017 nm. Overall, the variance of the bilayer was larger than that of the monolayer, which could be due to the monolayer measurements having a smaller sample size. The variance for 50 nN and 62.5 nN was not calculated since there were only two 50 nN measurements and a single 62.5 nN measurement.

As the optimal PeakForce$^{\mathrm{TM}}$ setpoint, 15 nN was selected, since the variance for 15 nN was among the lowest for both the monolayers and the bilayers. Additionally, the average height was close to its theoretical value for the bilayer and the closest for the monolayer.

The measurements with the PeakForce$^{\mathrm{TM}}$ setpoints 37.5 nN, 50 nN and 62.5 nN showed no damage on the tips, but on the 62.5 nN tip it was visible, that the tip was more worn than the 50 nN tip (Fig. 4.8(b), 4.9(b)). The height values increased when using these higher forces, showing that there was no layer compression for the bilayers.

A trilayer and a hexalayer were measured with different forces, ranging from 2.5 nN to 37.5 nN and from 2.5 nN to 50 nN, respectively, as well, to verify that there would still be no compression. As shown in Fig. 4.12, there did not seem to be any compression, when increasing the PeakForce$^{\mathrm{TM}}$ setpoint. For the hexalayer,

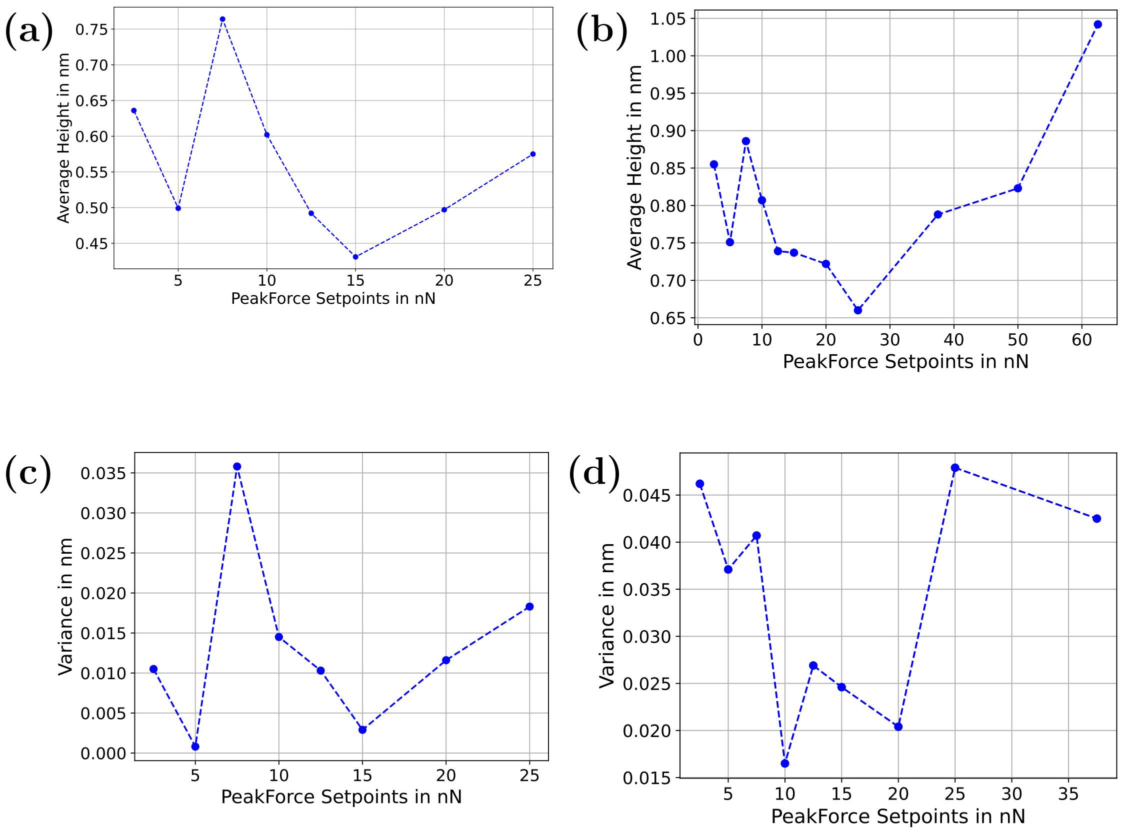


Figure 4.11: Mean height of monolayer (a) and bilayer (b) graphene across different PeakForce^{TM} setpoints; variation in height measurements for monolayer (c) and bilayer (d) graphene across different PeakForce^{TM} setpoints

a steady increase in height was observed from 20 nN onwards.

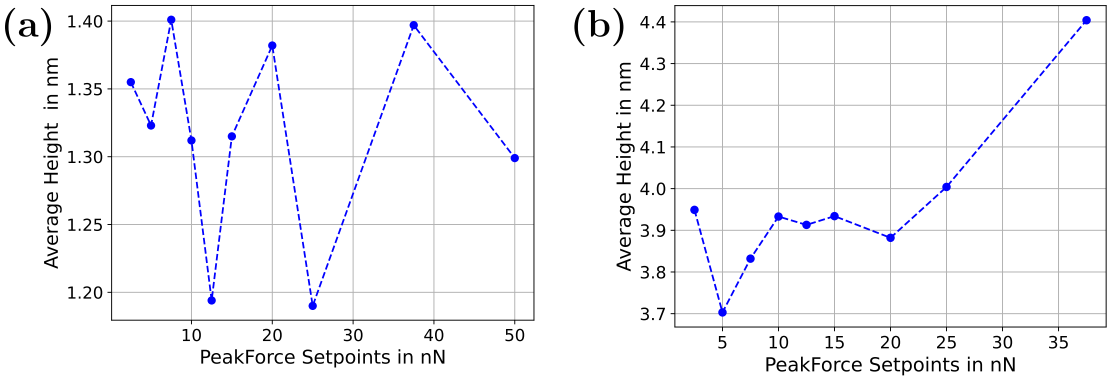


Figure 4.12: Height measurements for a trilayer (a) and a hexalayer (b) across various PeakForce^{TM} setpoints

4.3.5 Height Measurements of Graphene Layers

In total, including the flakes measured to determine the optimal PeakForce^{TM} setpoint, six monolayers, thirteen bilayers, four trilayers, four tetralayers, one pentalayer and one hexalayer were measured using a PeakForce^{TM} setpoint of 15 nN. For the mono-, bi-, tri-, tetra- and pentalayers, most of the measured heights agreed well with the theoretically expected height, which is a multiple of the theoretical height of the monolayer height 0.335 nm [25]. The mean height of the monolayers was 0.502 nm, while that of the bilayers was 0.699 nm. One measurement for the monolayer and three for the bilayer, however, were significantly far from the other values.

The outlier value of the measurement of the monolayer had a height of 0.857 nm, which could have caused it to be mischaracterized as a bilayer, while the other monolayers all ranged from 0.355 nm to 0.472 nm, which are not close to the theoretical value of the monolayer. Similarly, one bilayer outlier, with a height of 0.487 nm, could have been mischaracterized as a monolayer. Two additional bilayers, with a height of 0.892 nm and 0.918 nm, were close to the range of the trilayer height measurements, while the other measured values ranged from 0.598 nm to 0.773 nm.

The trilayers had a mean height of 1.084 nm. One of the four measured trilayers lying far above the other trilayers, with a height of 1.315 nm, which was closer to the tetralayer range. The other three trilayer heights ranged from 0.971 nm to 1.077 nm. The tetralayers had a mean height value of 1.209 nm, with three values lying close to each other, like in the trilayers, and one lying far below the others so that it fell more closely to the trilayers than the other tetralayers, with a height of 1.80 nm, while the others ranged from 1.338 nm to 1.34 nm.

The single pentalayer that was measured had a height of 1.812 nm and the single hexalayer measurement had a height of 3.934 nm.

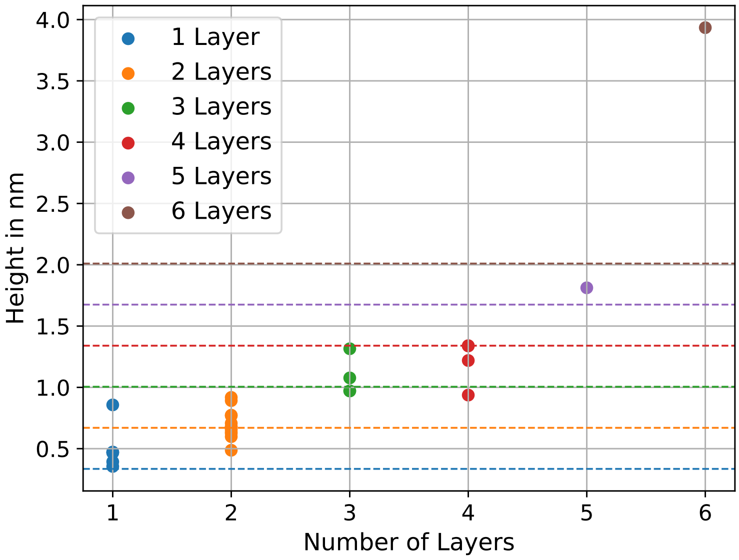


Figure 4.13: Height measurements performed with PeakForce^{TM} of every flake plotted depending on their number of layers, with the dashed lines in the corresponding colour to the number of layers representing the theoretical height.

When plotting the mean height of the layers, it is well visible, that from the monolayer to the pentalayer the height increased linearly, as shown in (Fig. 4.14). The hexalayer height value was far larger, than what would have been expected. When plotting the linear increase was 0.31±0.05 nm, which was close to the theoretical value of 0.335 nm [25].

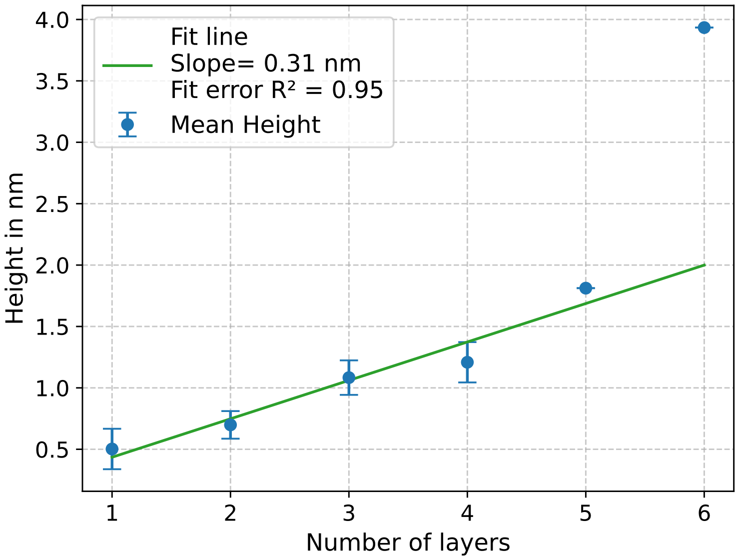


Figure 4.14: Mean height of every layer with their standard deviation plotted as error bars obtained with PeakForce^{TM} microscopy. The line is fitted for the mean values of the mono- to pentalayers excluding the hexalayer

4.3.6 Problems with PeakForce^{TM} Measurements

During these measurements, it was observed that not all flakes can reliably be measured with PeakForce^{TM} microscopy. For flakes with too many bubbles, for example (Fig. 4.15), it was not possible to accurately measure their height.

Inconsistent height measurements were observed at different positions on some flakes, despite no visible irregularities. Previous papers [25] had reported, that the overestimation of height using PeakForce^{TM} microscopy was due to a buffer layer between the graphene flake and the SiO_2, as shown in Fig. 4.16. Variations in buffer layer thickness across the flake could potentially also explain the discrepancies in the height measurements.

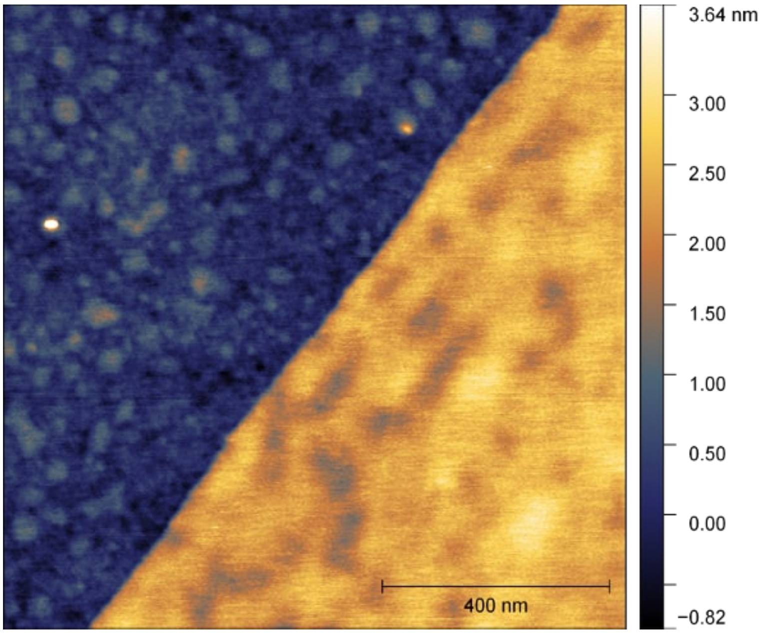


Figure 4.15: PeakForce^{TM} image of a graphene, where the height of the flake could not be measured due to the bubbles

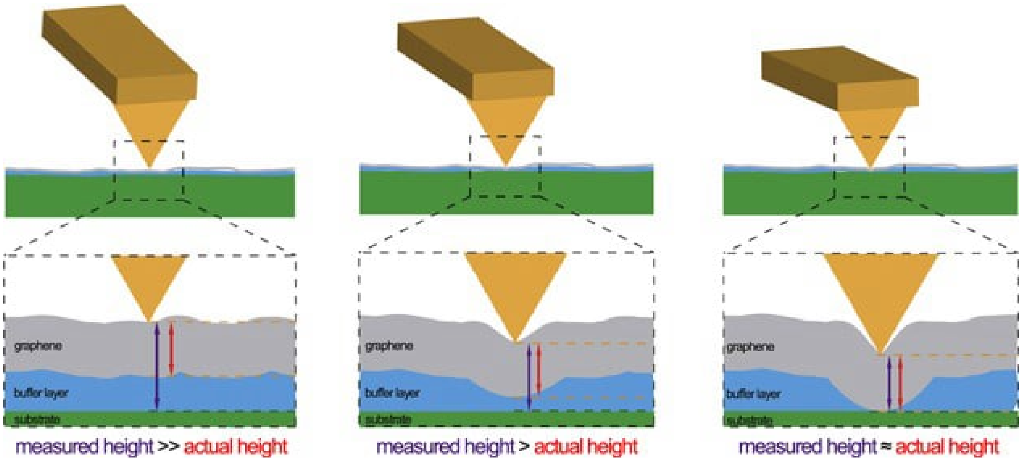


Figure 4.16: Compression of Buffer Layer during PeakForce^{TM} Measurement, taken from [25]

4.3.7 Multiple Height Measurements at Different Positions of the Flake

To determine if the number of layers of the seven graphene flakes, which had height measurements significantly deviating from the other height values of the same number of layers, could still be accurately identified, five additional PeakForce^{TM} scans were performed. The most consistent height across scans was taken as the true flake height.

All the seven flakes showed a significant deviation of height during this measurement. However, there were height values, that were relatively close to each other.

When taking the mean of these consistent values as the true height of the flake, of the seven flakes only two did not have height values, that were not close to the range of the other height values of the flakes that had the same number of layers as them.

The monolayer height measurements did not overlap with any other range any longer and had a range from 0.355 nm to 0.541 nm. For the bilayers, there was still a value that was far away from the other heights, which were measured, which ranged from 0.598 nm to 0.773 nm, at 0.940 nm. This values fell in the height range of the trilayers, which ranged from 0.920 nm to 1.077 nm. The trilayer values were well distinguishable from the tetralayers. The tetralayers had one value, which lay far away from the rest of the other three measurements, which ranged from 1.338 nm to 1.340 nm, at 1.830 nm. This value fell close to the measured pentalayer and also in the expected range of the pentalayer height.

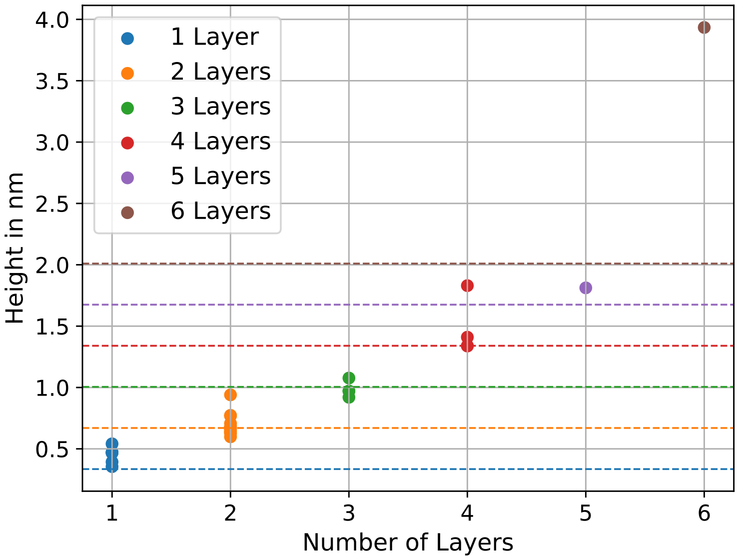


Figure 4.17: Height measurements of every flake obtained with PeakForce^{TM} plotted depending on their number of layers, where the height values of the flakes, that were measured five times at different places, replace the height that was previously measured for them. The dashed lines in the corresponding colour to the number of layers representing the theoretical height.

When replotting the mean height of each layer and fitting a line from the mono- to the pentalayer mean values, the linear increase was 0.35±0.03 nm, which agreed well with the theoretical value of 0.335 nm [25].

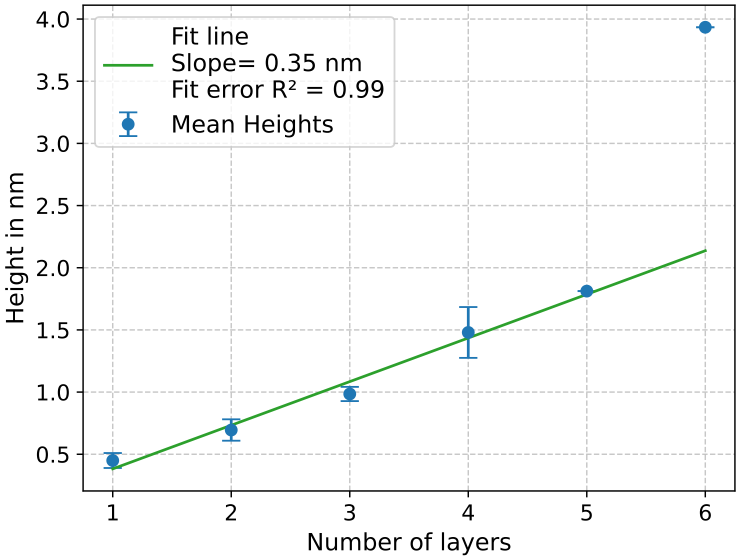


Figure 4.18: Mean height of every layer with their standard deviation plotted as error bars measured with PeakForce^{TM} microscopy. The line is fitted for the mean values of the mono- to pentalayers excluding the hexalayer and where the height values of the flakes, that were measured five times at different places, replace the height that was previously measured for them.

4.4 Comparison between Tapping and PeakForce^{TM} Mode

Both of the AFM modes took approximately the same time to image an area.

The height measurements taken in PeakForce^{TM} mode varied much less than those taken in tapping mode. Even when only a single measurement per flake was taken in PeakForce^{TM} mode, the standard deviations of the height measurements was about three times smaller than in tapping mode. This smaller range of the PeakForce^{TM} measurements also made it possible to differentiate between the number of layers more accurately, especially for samples consisting of more than one flake, since the heights overlapped significantly in tapping mode.

Further, the height measurements of graphene in PeakForce^{TM} mode agreed much better with the theoretical height than tapping mode, as shown in Fig. 4.20. This is highlighted by the line fitted through the mean values of the PeakForce^{TM} measurements, where only a single measurement was taken per flake, which had a slope of 0.31±0.04 nm, while the corresponding slope for tapping mode was 0.24±0.21 nm. Additionally, the error of the slope fit was larger for tapping mode $R^2 = 0.42$ than for PeakForce^{TM} $R^2 = 0.95$.

Even though, the PeakForce^{TM} measurements were more accurate and consistent, they still suffered from an overestimation and underestimation of the height like the tapping measurements, but to a lesser extent. The mean height deviated only from

Number of Layers	**Value**	**PeakForce^{TM} one (nm)**	**PeakForce^{TM} five (nm)**
1	Mean	0.50	0.45
	Std Dev	0.16	0.06
	Min	0.35	0.35
	Max	0.86	0.54
2	Mean	0.70	0.70
	Std Dev	0.11	0.09
	Min	0.49	0.60
	Max	0.92	0.94
3	Mean	1.08	0.98
	Std Dev	0.14	0.06
	Min	0.97	0.92
	Max	1.31	1.08
4	Mean	1.21	1.48
	Std Dev	0.16	0.20
	Min	0.94	1.34
	Max	1.34	1.83
5	Value	1.81	1.81
6	Value	3.93	3.93

Table 4.5: Mean value (Mean), standard deviation (Std Dev), and minimum (Min) and maximum (Max) measured heights using PeakForce^{TM}, where "PeakForce^{TM} one" are the values received by only measuring one spot on the flake and "PeakForce^{TM} five" by remeasuring certain values and using the average of the values of the five measurements of the flake performed on different locations

the theoretical height up to 0.17 nm for PeakForce^{TM}, while the mean heights of tapping mode deviated up to 0.83 nm. The PeakForce^{TM} mode seemed to be able to compress the buffer layer which had been speculated to be the cause of this overestimation better.

4.5 Comparison of Performed Measurements

An overview of all the mean values and standard deviations is given in Table 4.6. All the methods were able to characterize monolayer graphene well and most were also able to identify bilayers, but had difficulties differentiating between higher layer numbers.

The tapping mode height measurements were the most imprecise among the methods. Tapping mode overestimated the true height by a lot, yielding nearly double the height for monolayer, bilayer and trilayer samples, while also exhibiting large standard deviations, for example for the monolayer, it had a standard deviation of over 50% of the mean value. The height measurements obtained using tapping

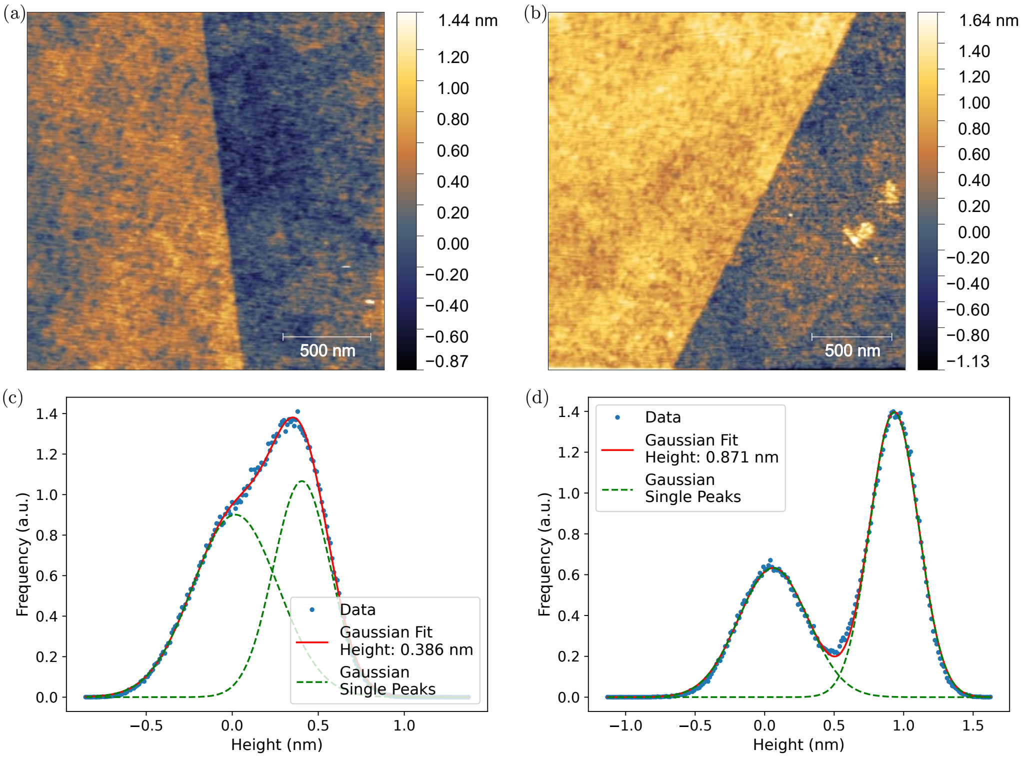


Figure 4.19: Scans of monolayer graphene taken with PeakForce$^{\mathrm{TM}}$ (a) and tapping mode (b) and the corresponding graph depicting the frequency each height was found with PeakForce$^{\mathrm{TM}}$ (c) and tapping mode (d)

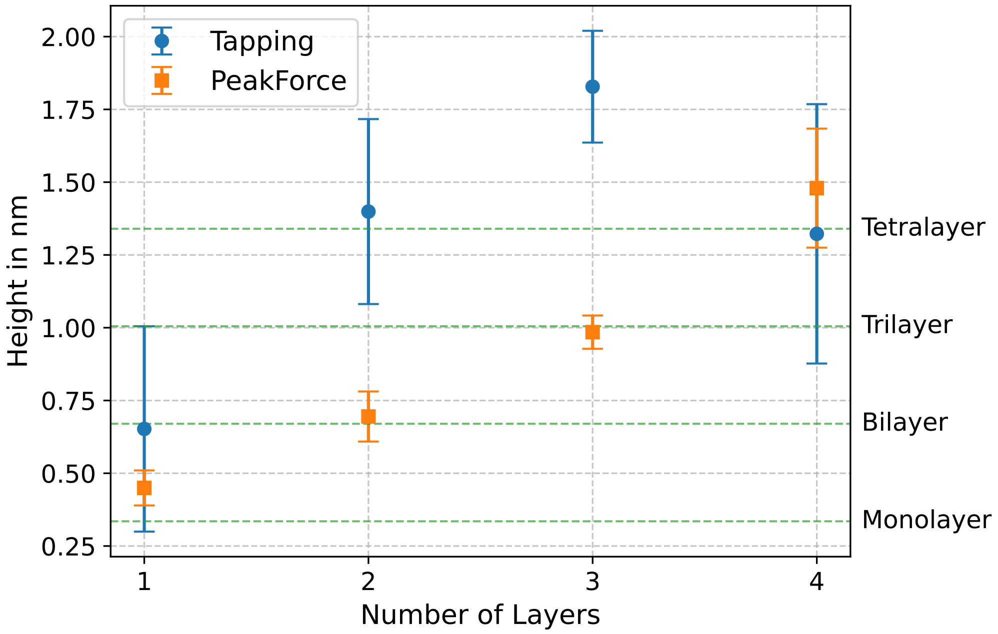


Figure 4.20: Mean height measured with tapping and PeakForce$^{\mathrm{TM}}$ mode with their standard deviations as error bars and green dashed lines representing the theoretically expected height of the different number of layers written on the right next to the line

mode were only able to characterize the monolayers.

The ratio of the G peak to the 2D peak was also unreliable and only able to characterize monolayers and bilayers with the mean ratio of trilayer to pentalayer graphene almost having identical values.

The FWHM measurements of the 2D peak were also able to characterize monolayers and bilayers reliably. The mean values of the FWHM were further apart than those of the ratio of the G peak to the 2D peak, even though the standard deviation ranges still overlapped for them. However, for pentalayers and hexalayers, the FWHM values were nearly identical when fitted with a single Lorentzian peak.

Like the FWHM, the position of the center of the 2D peak was able to reliably identify monolayers and bilayers. For the fitting method using the single Lorentzian peak, the trilayer standard deviation also did not overlap with any other standard deviations. However, when looking at the range of the measured values, the range of the bilayers and monolayers already started to overlap for the fitting methods using a single Lorentzian peak and the bilayer and trilayer range overlapped as well for both fitting methods (see Table 4.2). The standard deviations for the pentalayers and hexalayers were also rather larger, to the extent that the ranges of the standard deviations still overlapped with the tetralayers, and for the fitting method using multiple Lorentzian peaks also with the trilayers, even though the mean values of the pentalayers and hexalayers were farther apart than those of the FWHM.

The PeakForce$^{\text{TM}}$ height measurements mean values showed a clear trend, increasing steadily with the number of layers up to and including pentalayers. However, although the standard deviation ranges of only the monolayers and bilayers did not overlap in the PeakForce$^{\text{TM}}$ measurements performed at a single spot on the sample, the individual measured values did. The PeakForce$^{\text{TM}}$ measurements, where a few flakes were measured in five different spots, were able to characterize the number of layers up to tetralayers reliably.

Method	**Mono-Layer**	**Bi-Layer**	**Tri-Layer**	**Tetra-Layer**	**Penta-Layer**	**Hexa-Layer**
Tapping Mode in nm	**0.65 ± 0.35**	1.40 ± 0.32	1.83 ± 0.19	1.32 ± 0.45	-	-
PeakForce$^{\text{TM}}$ one in nm	0.50 ± 0.16	0.70 ± 0.11	1.08 ± 0.14	1.21 ± 0.16	1.81*	3.93*
PeakForce$^{\text{TM}}$ five in nm	**0.45 ± 0.06**	**0.70 ± 0.09**	**0.98 ± 0.06**	**1.48 ± 0.20**	1.81*	3.93*
FWHM single in cm^{-1}	**26.6 ± 1.3**	**43.3 ± 0.5**	55.3 ± 1.5	57.4 ± 1.7	59.9 ± 0.8	59.7 ± 0.7
FWHM multi in cm^{-1}	-	**51.7 ± 2.2**	63.1 ± 1.6	64.8 ± 2.6	65.1 ± 1.0	66.6 ± 1.5
2D center single in cm^{-1}	**2664.6 ± 4.6**	**2686.7 ± 2.8**	**2692.6 ± 2.4**	2697.3 ± 0.7	2698.9 ± 4.5	2702.4 ± 4.3
2D center multi in cm^{-1}	-	**2686.7 ± 2.7**	2694.0 ± 1.6	2697.5 ± 1.0	2697.5 ± 4.0	2699.2 ± 3.8
G/2D ratio single	**0.42 ± 0.08**	**0.72 ± 0.09**	0.96 ± 0.19	0.97 ± 0.45	0.97 ± 0.45	1.18 ± 0.45
G/2D ratio multi	-	**0.77 ± 0.10**	1.09 ± 0.02	1.07 ± 0.12	1.08 ± 0.10	1.31 ± 0.13

Table 4.6: Mean values with their standard deviations for all the methods used to characterize the number of graphene layers: Tapping Mode, PeakForce$^{\text{TM}}$ measuring a single spot of the flake (PeakForce$^{\text{TM}}$ one), PeakForce$^{\text{TM}}$ measuring selected flakes in five different locations and averaging of the three values lying closest to each other (PeakForce$^{\text{TM}}$ five), FWHM of the 2D peak (FWHM) measured with Raman, center of the 2D peak measured with Raman (2D center), ratio of the G peak to the 2D peak (G/2D ratio) measured with Raman, with “single” and “multi” denoting whether the 2D peak is fitted with a single Lorentzian peak or with multiple Lorentzian peaks fitted depending on the number of layers
*no standard deviation as only one sample was measured
Values for which the results, considering the standard deviation, do not overlap with those of other number of layers determined with the same method are highlighted in bold

Chapter 5

Conclusions and Outlook

This thesis confirms that the AFM mode, PeakForce$^{\mathrm{TM}}$, is a viable method to determine the number of layers of graphene flakes with high accuracy, especially when measuring the flake at five different locations, of the 29 measured graphene samples, 27 were consistent with the layer characterization determined by optical contrast. In general, the measurements agreed well with the theoretically expected height, with the mean values only deviating up to 0.17 nm from it for samples up to pentalayers. In contrast, tapping mode exhibited larger deviations from the theoretically expected height by as much as 0.83 nm.

As already observed in [25], the buffer layer between the graphene and the substrate leads to an overestimation of the height measured with PeakForce$^{\mathrm{TM}}$ and especially with tapping mode. Using PeakForce$^{\mathrm{TM}}$, variations in measured heights across different positions of the same sample were noticed as well.

Furthermore, it was observed that by using Raman spectroscopy, it is difficult to reliably differentiate between graphene flakes that consist of three or more layers and therefore, additional methods should be used to confirm the number of layers. However, Raman spectroscopy is able to reliably distinguish monolayer and bilayer graphene flakes, from each other and from samples with three or more layers, when analyzing the FWHM and the center of the 2D peak.

Interestingly, while measuring at different PeakForce$^{\mathrm{TM}}$ setpoints, it was observed that the measured height began to increase after a certain force threshold. The underlying cause of this effect remains unclear and warrants further research. In this thesis, the different PeakForce$^{\mathrm{TM}}$ setpoints were mainly explored on monolayers and bilayers. The setpoint of 15 nN had the mean value that was closest to the theoretical height of the monolayers, and 25 nN was the closest to the theoretical height of the bilayers. Since the force required to obtain the mean value closest to the theoretical value increased from monolayer to bilayer, it might further increase for a higher number of layers. This could be a topic of further research, in order to obtain more accurate height measurements and thus more reliable classifications of the number of layers especially for a higher number of layers.

PeakForce^{TM} microscopy appears to be a promising technique to determine the number of layers of other 2D materials with similar structure. Further research would be helpful to verify this hypothesis.

Scanning a larger area with PeakForce^{TM} microscopy might potentially even out the height differences caused by the buffer layer and lead to a more reliable identification of the number of layers, and might be subject for futher investigation. A better exfoliation process could minimize the buffer layer and lead to more accurate height measurements for PeakForce^{TM} as well as tapping mode.

For the PeakForce^{TM} measurements, the auto control was used for the “Gain”, “Scan Rate” and “Z Limit”, see Fig. A.6. Further experiments could manually vary those values to investigate whether they lead to an improved scan quality.

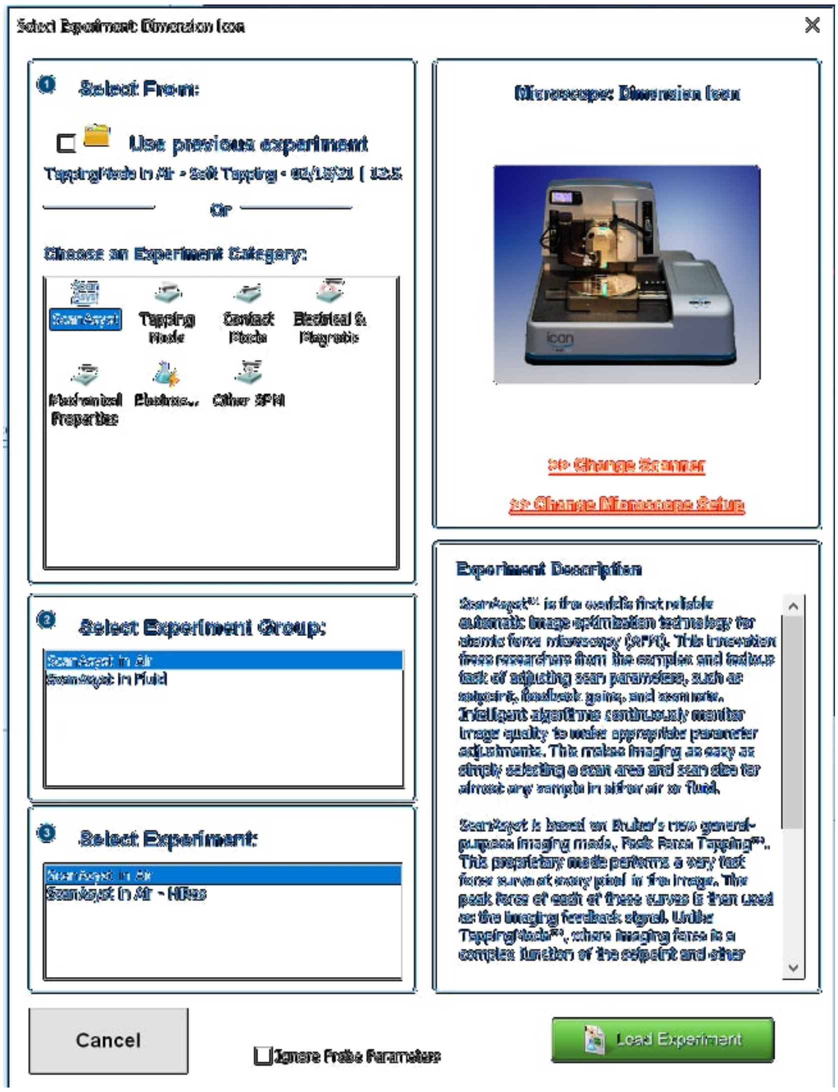
Appendix A

PeakForce Operating Procedure

This chapter provides a guide to perform PeakForce microscopy using Bruker Dimension Icon Atomic Force Microscope and the software NanoScope 6 controller in the ScanAssist in Air mode, starting from the set up and going to the thermal tune to measure the spring constant and the deflection sensitivity to the scanning process.

A.1 AFM Setup

1. Start the software and select the experiment category “ScanAsyst” with the experiment group and experiment “ScanAsyst in Air”, see Fig. A.1
2. Attach a cantilever to the probe holder (Fig. A.2) by
    (a) pulling back the spring-loaded clip,
    (b) placing the cantilever into the cantilever mounting groove, so that it is straight and touches the back edge of the groove,
    (c) pushing the clip forward to secure the cantilever
3. Install the probe holder by
    (a) loosening the dovetail release screw,
    (b) pulling up the SPM scanner,
    (c) attaching the probe holder to the bottom of the SPM scanner with the four pins,
    (d) sliding the SPM scanner back down,
    (e) fastening the dovetail release screw
4. Press “Select Probe” and choose the correct Cantilever, see Fig. A.4



ScanAsyst™ is the world's first reliable automatic image optimization technology for atomic force microscopy (AFM). This innovation frees researchers from the complex and tedious task of adjusting scan parameters, such as setpoint, feedback gains, and scan rate. Intelligent algorithms continuously monitor image quality to make appropriate parameter adjustments. This makes imaging as easy as simply selecting a scan area and scan size for almost any sample in either air or fluid.

ScanAsyst is based on Bruker's new general-purpose imaging mode, Peak Force Tapping™. This proprietary mode performs a very fast force curve at every pixel in the image. The peak force of each of these curves is then used as the imaging feedback signal. Unlike TappingMode™, where imaging force is a complex function of the setpoint and other

Figure A.1: Experiment Selected for the PeakForce Measurements

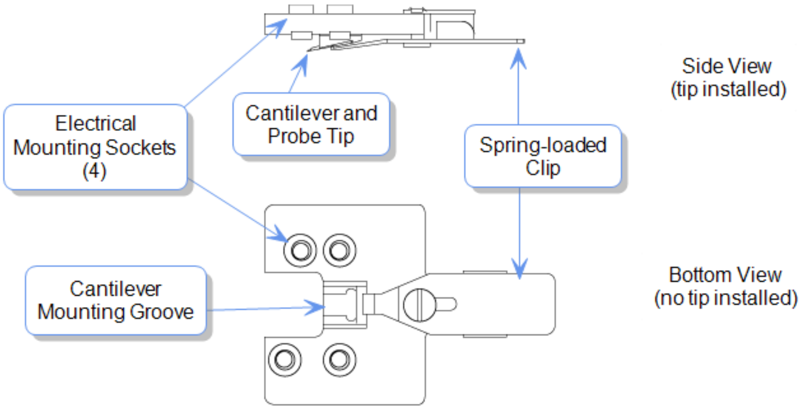


Figure A.2: Schematic drawing of Probe Holder, taken from [66]

5. Adjust the laser using the laser control knobs, so that it is pointed at the tip of the cantilever, see Fig A.4

6. Tilt the photodetector with the photodetector adjustment knobs until the

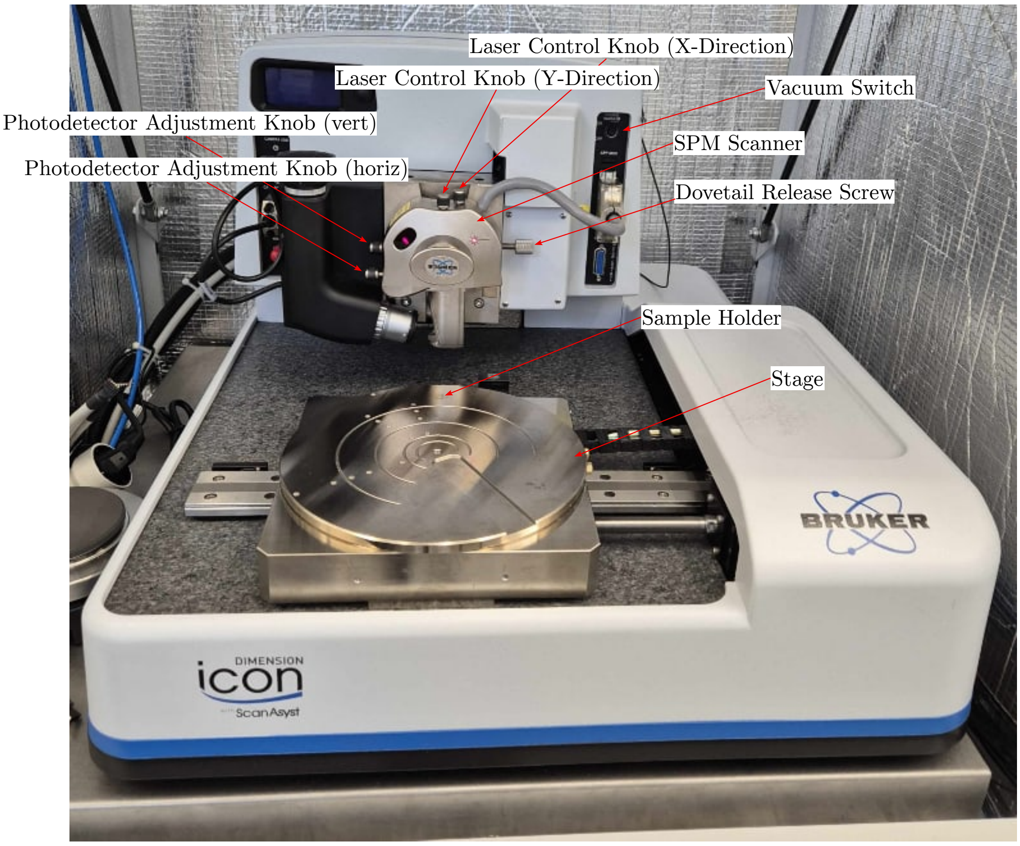


Figure A.3: Picture of Bruker Dimension Icon Atomic Force Microscope

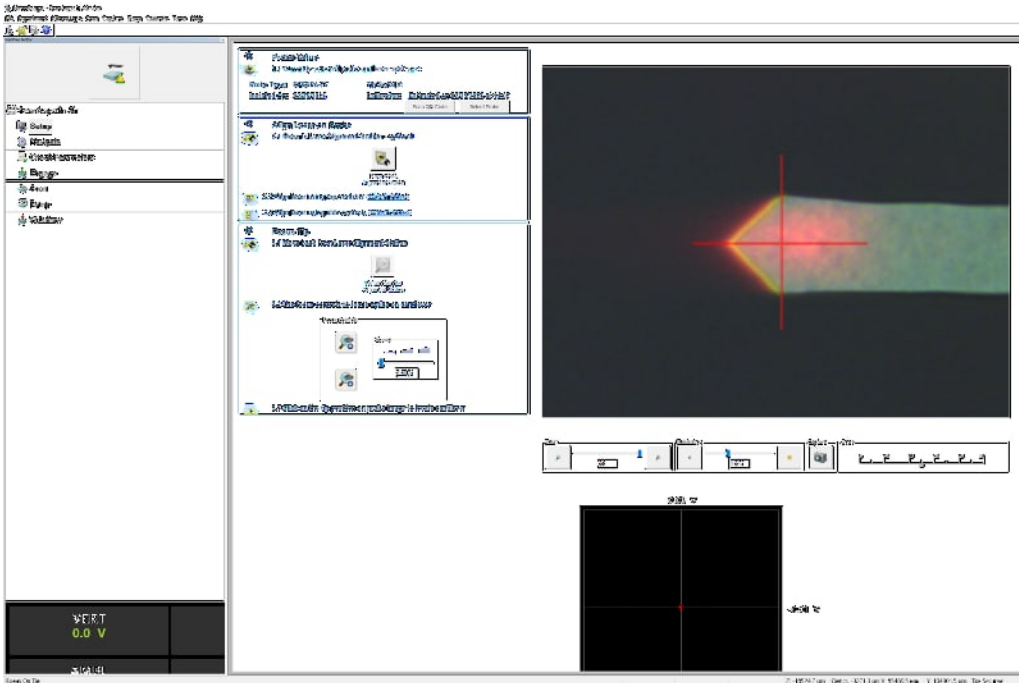


Figure A.4: Setup Screen

laser is in the middle of the photodetector. This is achieved, when the laser signal values are 0 for both the vertical and horizontal components.

7. Focus the cantilever using the focus controls to move the optics up and down
8. Place the red crosshair on the tip by clicking on the tip position on the live camera feed
9. Place the sample on the vacuum chuck and turn it on
10. Click the “Navigate” icon and move the stage so that the sample is underneath the cantilever using the “XY Control” buttons
11. Focus the sample by moving the stage up and down using the “Scan Head” buttons
12. Use the “XY Control” buttons to navigate to the desired measurement location
13. Click the “Check Parameters” icon to move on to the thermal tuning

A.2 Thermal Tune

14. Expand the toolbar by pressing “Expanded Mode”
15. Click on “Thermal Tune” in the toolbar to open the Thermal Tune window
16. Select the smallest frequency range in which the frequency of the used cantilever is contained
17. Click on “Acquire Data” to record the deflection signal for about 30 seconds [67]
18. Select “Lorentzian (Air)”
19. Click on “PSD Bin Width” to reduce the noise
20. Drag markers from the side of the plot to the sides of the resonance peak, see Fig. A.5
21. Click “Fit Data” to fit the Lorentzian peak to the resonance peak
22. Click on “K by Thermal” and “Defl Sens by Thermal” and accept the values
23. Repeat steps 17 to 22 to ensure a more accurate determination of the spring constant (see chapter 4.3.6) and click on “Return and Save Changes”

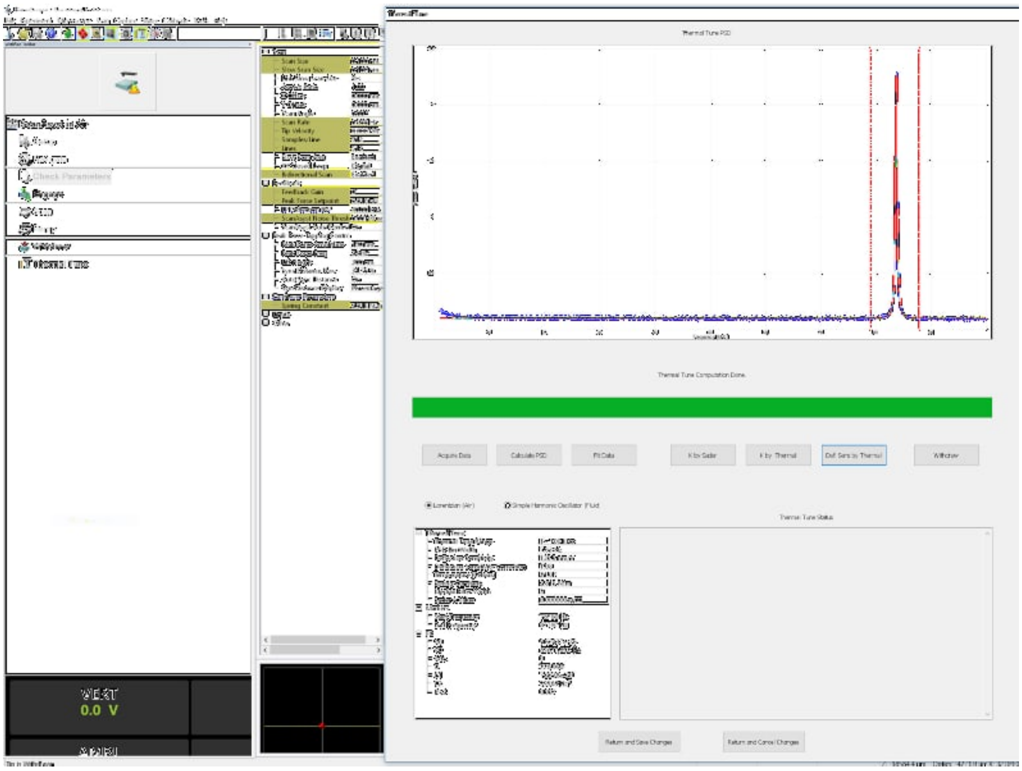


Figure A.5: Thermal Tuning Process

A.3 Initiating Scan

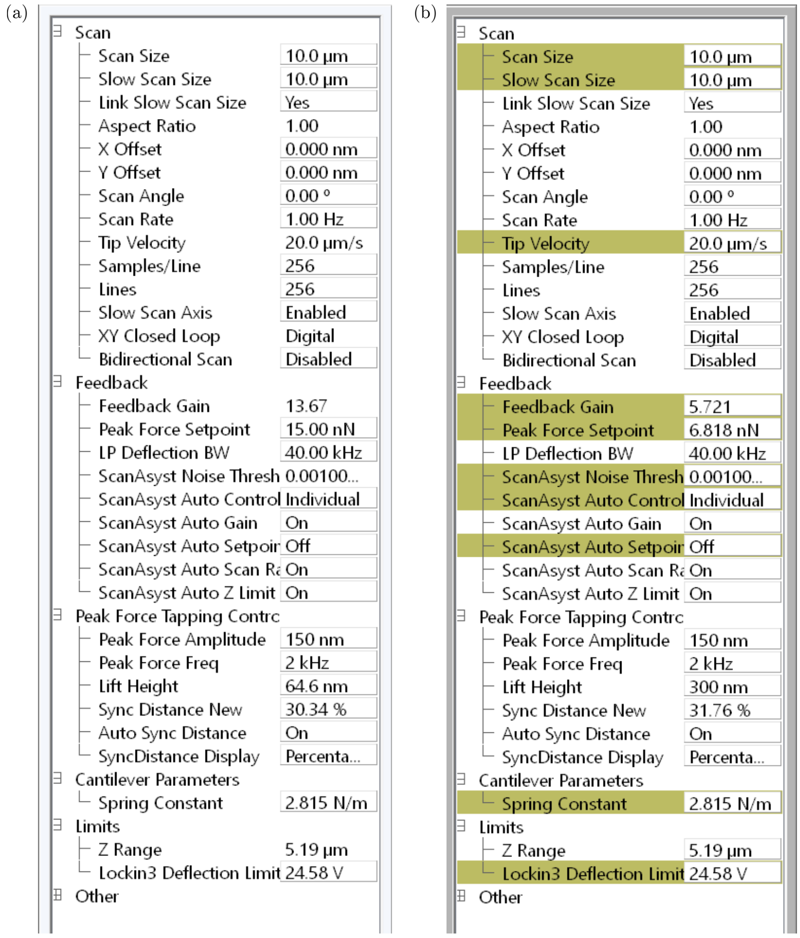
24. Click on “Check Parameters”

25. Change the “Scan Size” and the “Sample/Line” to the desired size and resolution of the scan

26. Set “ScanAsyst Auto Control” to “Individual” and turn off “ScanAsyst Auto Setpoint”

27. Change the setpoint to a value between 2.5 nN and 10 nN, since a too small setpoint can lead to difficulties when engaging, and a too high setpoint might damage the tip due to the setpoint being automatically increased by the software during the engaging process, sometimes to such a high force as to break the cantilever.

28. Click on “Engage” to lower the cantilever to the scanning position and begin the scan. If the spring constant is determined incorrectly, engagement issues may arise, as the program either totally fails to engage or incorrectly determines the “Lift Height” or the “Peak Force Amplitude”, which are automatically set when engaging. This is noticeable, when the Force-Time curve does not exhibit a well-defined up and down peak, as shown in Fig. A.7, but



	(a)	(b)
Scan		
Scan Size	10.0 µm	10.0 µm
Slow Scan Size	10.0 µm	10.0 µm
Link Slow Scan Size	Yes	Yes
Aspect Ratio	1.00	1.00
X Offset	0.000 nm	0.000 nm
Y Offset	0.000 nm	0.000 nm
Scan Angle	0.00 °	0.00 °
Scan Rate	1.00 Hz	1.00 Hz
Tip Velocity	20.0 µm/s	20.0 µm/s
Samples/Line	256	256
Lines	256	256
Slow Scan Axis	Enabled	Enabled
XY Closed Loop	Digital	Digital
Bidirectional Scan	Disabled	Disabled
Feedback		
Feedback Gain	13.67	5.721
Peak Force Setpoint	15.00 nN	6.818 nN
LP Deflection BW	40.00 kHz	40.00 kHz
ScanAsyst Noise Thresh	0.00100...	0.00100...
ScanAsyst Auto Control	Individual	Individual
ScanAsyst Auto Gain	On	On
ScanAsyst Auto Setpoin	Off	Off
ScanAsyst Auto Scan R	On	On
ScanAsyst Auto Z Limit	On	On
Peak Force Tapping Contro		
Peak Force Amplitude	150 nm	150 nm
Peak Force Freq	2 kHz	2 kHz
Lift Height	64.6 nm	300 nm
Sync Distance New	30.34 %	31.76 %
Auto Sync Distance	On	On
SyncDistance Display	Percenta...	Percenta...
Cantilever Parameters		
Spring Constant	2.815 N/m	2.815 N/m
Limits		
Z Range	5.19 µm	5.19 µm
Lockin3 Deflection Limit	24.58 V	24.58 V
Other		

Figure A.6: Example parameters for PeakForce microscopy (a) before engaging and (b) during the scanning process

instead resembles something more similar to a sinusoidal function or when the small fluctuations outside this peak become wider or have a large amplitude. Performing step 23 generally prevents these issues.

29. Once the engaging process is complete, adjust the setpoint to the desired value and examine the Force-Time curve to confirm a successful engagement. If the engagement is not successful, manually adjust the “Lift Height” and “Peak Force Amplitude” or repeat the engagement process by clicking “Withdraw” to retract the tip.

A.4 Removing the Sample

30. Click on “Withdraw” to stop the measurement and lift the tip

31. Click on the “Navigate” icon and click on “Sample Load Position” to move the stage and unload the sample

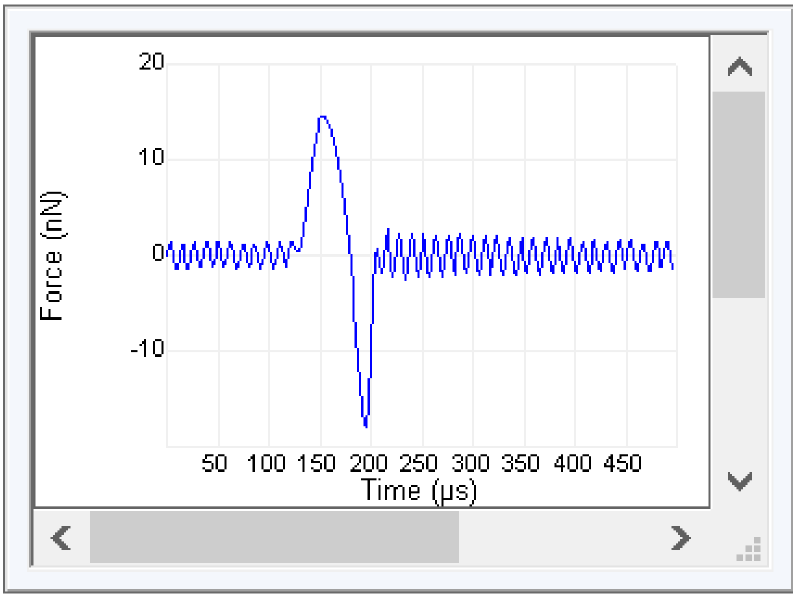


Figure A.7: Force-Time Curve

Appendix B

Python Scripts for Raman Spectroscopy

B.1 Python Script for the 2D and G Raman Peak Fit Using a Single Lorentzian Peak

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit
import os
import pandas as pd

# Define Lorentzian function
def Lorentzian(x, amp, cen, wid,y0):
    return (y0+2*amp*wid**2/(4*(x-cen)**2+wid**2))

# Define integral of Lorentzian function
def integral(x1, x2, amp, cen, wid):
    return (amp*wid*np.arctan((2*x1-2*cen)/wid)-amp*wid*np.arctan((2*x2-2*cen)/wid))

# Range of values used to fit for 2D-peak for the G-peak x_min = 1480 and x_max = 1680 were used
x_min = 2200
x_max = 3200

# Initialize list for storing file names and fit values
results = []

file_list = [...]

for file_path in file_list:
    try:
        # Import data
        file_name = os.path.splitext(os.path.basename(file_path))[0]
        data = np.loadtxt(f'raman graphene/{file_name}.txt', delimiter='\t')
        x_data = data[:, 0]
        y_data = data[:, 1]

        # Select only the data within the specified x range
        fit_mask = (x_data >= x_min) & (x_data <= x_max)
```

```
        x_fit = x_data[fit_mask]
        y_fit = y_data[fit_mask]

        # Define boundary parameters
        # [min: amp, cen, wid, y0], [max: amp, cen, wid, y0]
        bounds = ([30, 2640, 20, min(y_fit)], [np.inf, 2750, 100, max(y_fit)])

        parameters, covariance = curve_fit(Lorentzian, x_fit, y_fit, bounds=bounds)

        fit_amp = parameters[0]
        fit_cen = parameters[1]
        fit_wid = parameters[2]
        fit_y0 = parameters[3]

        # Calculate integral
        intint=integral(x_max, x_min, fit_amp, fit_cen, fit_wid)

        print(f'{file_name}: {fit_amp:.2f}, {fit_cen:.2f}, {fit_wid:.2f},
 {fit_y0:.2f}, integrated intensity {intint:.2f}')

        # Define the folder path
        output_folder = '...'

        # Append file name and amplitude to results
        results.append((file_name, fit_wid, fit_cen, fit_amp,intint))

        fit_y = Lorentzian(x_fit, fit_amp, fit_cen, fit_wid, fit_y0)
        plt.plot(x_fit, y_fit, '.', label='Measured Raman Data')
        plt.plot(x_fit, fit_y, 'r-', label=f'Lorentzian Fit')
        plt.ylabel("Intensity (cts)")
        plt.xlabel("Raman Shift (cm$^{-1}$)")
        plt.legend(loc='upper right')
        plt.legend()
        plt.show()
        plt.close()

    except Exception as e:
        print(f"Error processing file {file_path}: {e}")

# Create a table
results_df = pd.DataFrame(results, columns=['File Name', 'FWHM', 'Center', 'Amplitude', 'integrated intensity'])

# Save the results table as a CSV file
 results_df.to_csv(FWHM1.csv', index=False)
```

B.2 Python Script for the 2D Raman Peak Fit Using Multiple Lorentzian Peaks

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit
import os
import pandas as pd

# Initialize list for storing file names and amplitudes
results = []
```

```
# Define single Lorentzian function
def Lorentzian_single(x, amp, cen, wid, y0):
    return y0 + 2 * amp * wid**2 / (4 * (x - cen)**2 + wid**2)

def integral(x1, x2, amp, cen, wid):
    return (amp*wid*np.arctan((2*x1-2*cen)/wid)-amp*wid*np.arctan((2*x2-2*cen)/wid))

# Define multi-Lorentzian function for 3 Lorentzians
def Lorentzian_multi_3(x, amp1, cen1, wid1, y01, amp2, cen2, wid2, y02, amp3, cen3, wid3, y03):
    return (y01 + 2 * amp1 * wid1**2 / (4 * (x - cen1)**2 + wid1**2) +
        2 * amp2 * wid2**2 / (4 * (x - cen2)**2 + wid2**2) +
        2 * amp3 * wid3**2 / (4 * (x - cen3)**2 + wid3**2) )

def integral_3(x1, x2, amp1, cen1, wid1, y01, amp2, cen2, wid2, y02, amp3, cen3, wid3, y03):
    return (amp1*wid1*np.arctan((2*x1-2*cen1)/wid1)-amp1*wid1*np.arctan((2*x2-2*cen1)/wid1)+
            amp2*wid2*np.arctan((2*x1-2*cen2)/wid2)-amp2*wid2*np.arctan((2*x2-2*cen2)/wid2)+
            amp3*wid3*np.arctan((2*x1-2*cen3)/wid3)-amp3*wid3*np.arctan((2*x2-2*cen3)/wid3))


# Define multi-Lorentzian function for 4 Lorentzians
def Lorentzian_multi_4(x, amp1, cen1, wid1, y01, amp2, cen2, wid2, y02,
 amp3, cen3, wid3, y03, amp4, cen4, wid4, y04):
    return (y01 + 2 * amp1 * wid1**2 / (4 * (x - cen1)**2 + wid1**2) +
        2 * amp2 * wid2**2 / (4 * (x - cen2)**2 + wid2**2) +
        2 * amp3 * wid3**2 / (4 * (x - cen3)**2 + wid3**2) +
        2 * amp4 * wid4**2 / (4 * (x - cen4)**2 + wid4**2))

def integral_4(x1, x2, amp1, cen1, wid1, y01, amp2, cen2, wid2, y02,
 amp3, cen3, wid3, y03, amp4, cen4, wid4, y04):
    return (amp1*wid1*np.arctan((2*x1-2*cen1)/wid1)-amp1*wid1*np.arctan((2*x2-2*cen1)/wid1)+
            amp2*wid2*np.arctan((2*x1-2*cen2)/wid2)-amp2*wid2*np.arctan((2*x2-2*cen2)/wid2)+
            amp3*wid3*np.arctan((2*x1-2*cen3)/wid3)-amp3*wid3*np.arctan((2*x2-2*cen3)/wid3)+
            amp4*wid4*np.arctan((2*x1-2*cen4)/wid4)-amp4*wid4*np.arctan((2*x2-2*cen4)/wid4))

# Define multi-Lorentzian function for 6 Lorentzians
def Lorentzian_multi_6(x, amp1, cen1, wid1, y01, amp2, cen2, wid2, y02, amp3, cen3, wid3, y03,
 amp4, cen4, wid4, y04,  amp5, cen5, wid5, y05, amp6, cen6, wid6, y06):
    return (y01 + 2 * amp1 * wid1**2 / (4 * (x - cen1)**2 + wid1**2) +
        2 * amp2 * wid2**2 / (4 * (x - cen2)**2 + wid2**2) +
        2 * amp3 * wid3**2 / (4 * (x - cen3)**2 + wid3**2) +
        2 * amp4 * wid4**2 / (4 * (x - cen4)**2 + wid4**2) +
        2 * amp5 * wid5**2 / (4 * (x - cen5)**2 + wid5**2) +
        2 * amp6 * wid6**2 / (4 * (x - cen6)**2 + wid6**2))

def integral_6(x1, x2, amp1, cen1, wid1, y01, amp2, cen2, wid2, y02, amp3, cen3, wid3, y03,
 amp4, cen4, wid4, y04, amp5, cen5, wid5, y05, amp6, cen6, wid6, y06):
    return (amp1*wid1*np.arctan((2*x1-2*cen1)/wid1)-amp1*wid1*np.arctan((2*x2-2*cen1)/wid1)+
            amp2*wid2*np.arctan((2*x1-2*cen2)/wid2)-amp2*wid2*np.arctan((2*x2-2*cen2)/wid2)+
            amp3*wid3*np.arctan((2*x1-2*cen3)/wid3)-amp3*wid3*np.arctan((2*x2-2*cen3)/wid3)+
            amp4*wid4*np.arctan((2*x1-2*cen4)/wid4)-amp4*wid4*np.arctan((2*x2-2*cen4)/wid4)+
            amp5*wid5*np.arctan((2*x1-2*cen5)/wid5)-amp5*wid5*np.arctan((2*x2-2*cen5)/wid5)+
            amp6*wid6*np.arctan((2*x1-2*cen6)/wid6)-amp6*wid6*np.arctan((2*x2-2*cen6)/wid6))

# Range of values used to fit
x_min = 2200
x_max = 3200

# List of files
```

```
file_list = [...]

for file_path in file_list:
    try:
        # Import data
        file_name = os.path.splitext(os.path.basename(file_path))[0]
        data = np.loadtxt(f’raman graphene/{file_name}.txt’, delimiter=’\t’)
        x_data = data[:, 0]
        y_data = data[:, 1]

        # Select only the data within the specified x range
        fit_mask = (x_data >= x_min) & (x_data <= x_max)
        x_fit = x_data[fit_mask]
        y_fit = y_data[fit_mask]

        # Determine the parameter set based on the file type
        if ’_m’ in file_name:
            # Fit using single Lorentzian
            bounds = ([30, 2650, 20, min(y_fit)], [np.inf, 2750, 100, max(y_fit)])
            parameters, covariance = curve_fit(Lorentzian_single, x_fit, y_fit, bounds=bounds)
            fit_amp, fit_cen, fit_wid, fit_y0 = parameters

            # Generate fit data for plotting
            fit_y = Lorentzian_single(x_fit, fit_amp, fit_cen, fit_wid, fit_y0)

            intint=integral(x_max, x_min, fit_amp, fit_cen, fit_wid)

            print(f’{file_name}: {fit_amp:.2f}, {fit_cen:.2f}, {fit_wid:.2f}, {fit_y0:.2f}, integrated intensity {intint:.2f

            center=fit_cen
            y01=fit_y0

            # Append file name and amplitude to results
            layers=1
            results.append((layers, file_name, fit_amp, fit_wid, fit_cen, intint))

        elif ’_b’ in file_name:
            # Fit using multi-Lorentzian for 4 peaks
            # amp, cen, wid, y
            bounds = ([10, 2600, 15, 1250] * 4, [100, 2775, 100, 1350] * 4)
            parameters, covariance = curve_fit(Lorentzian_multi_4, x_fit, y_fit, bounds=bounds)
            (amp1, cen1, wid1, y01, amp2, cen2, wid2, y02,
             amp3, cen3, wid3, y03, amp4, cen4, wid4, y04) = parameters

            # Generate the combined fit
            x_dense = np.linspace(x_min, x_max, 1000)
            combined_y = Lorentzian_multi_4(x_dense, *parameters)

            # Find the max value of the combined curve
            max_intensity = np.max(combined_y)-y01

            # Calculate half of this max value
            half_max = max_intensity / 2 + y01

            # Append file name and amplitude to results
            #results.append((file_name, max_intensity))

            layers=2
```

```
            # Find the indices where combined_y is closest to half_max
            try:
                # Get the indices of the two points closest to half_max
                indices = np.where(np.diff(np.sign(combined_y - half_max)))[0]

                if len(indices) >= 2:
                    # Extract the x-values at these indices
                    x_left = x_dense[indices[0]]
                    x_right = x_dense[indices[-1]]

                    # Calculate overall FWHM as the difference between these points
                    overall_FWHM = abs(x_right - x_left)
                    center=x_left+(overall_FWHM/2)
                    intint=integral_4(x_max, x_min, *parameters)
                    results.append((layers, file_name, max_intensity, overall_FWHM, center, intint))
                    print(f'{file_name}, x_left: {x_left:.2f}, x_right: {x_right:.2f}, FWHM: {overall_FWHM:.2f},
cen: {center:.2f}, intensity: {max_intensity:.2f}, y: {y01:.2f}, integral: {intint:.2f}')
                else:
                    print(f'{file_name}: Less than two half-max crossings found.')
                    overall_FWHM = np.nan

            except Exception as e:
                print(f"Error calculating FWHM for {file_name}: {e}")
                overall_FWHM = np.nan

            # Generate fit data for plotting
            fit_y = Lorentzian_multi_4(x_fit, *parameters)

        elif '_t' in file_name:
            # Fit using multi-Lorentzian for 6 peaks
            # amp, cen, wid, y
            bounds = ([10, 2600, 15, 1250] * 6, [np.inf, 2775, 100, 1350] * 6)
            parameters, covariance = curve_fit(Lorentzian_multi_6, x_fit, y_fit, bounds=bounds)
            (amp1, cen1, wid1, y01, amp2, cen2, wid2, y02,
             amp3, cen3, wid3, y03, amp4, cen4, wid4, y04,
             amp5, cen5, wid5, y05, amp6, cen6, wid6, y06) = parameters

            # Generate the combined fit
            x_dense = np.linspace(x_min, x_max, 1000)
            combined_y = Lorentzian_multi_6(x_dense, *parameters)

            # Find the max value of the combined curve
            max_intensity = np.max(combined_y)-y01

            # Calculate half of this max value
            half_max = max_intensity / 2 + y01

            layers=3

            # Append file name and amplitude to results
            #results.append((file_name, max_intensity))

            # Find the indices where combined_y is closest to half_max
            try:
                # Get the indices of the two points closest to half_max
                indices = np.where(np.diff(np.sign(combined_y - half_max)))[0]

                if len(indices) >= 2:
                    # Extract the x-values at these indices
```

```
                    x_left = x_dense[indices[0]]
                    x_right = x_dense[indices[-1]]

                    # Calculate overall FWHM as the difference between these points
                    overall_FWHM = abs(x_right - x_left)
                    center=x_left+(overall_FWHM/2)
                    intint=integral_6(x_max, x_min, *parameters)
                    results.append((layers, file_name, max_intensity, overall_FWHM, center, intint))
                    print(f'{file_name}, x_left: {x_left:.2f}, x_right: {x_right:.2f}, FWHM: {overall_FWHM:.2f},
cen: {center:.2f}, intensity: {max_intensity:.2f}, y: {y01:.2f}, integral: {intint:.2f}')
                else:
                    print(f'{file_name}: Less than two half-max crossings found.')
                    overall_FWHM = np.nan

            except Exception as e:
                print(f"Error calculating FWHM for {file_name}: {e}")
                overall_FWHM = np.nan

            # Generate fit data for plotting
            fit_y = Lorentzian_multi_6(x_fit, *parameters)

        elif '_4' in file_name or '_5' in file_name or '_6' in file_name:
            # Fit using multi-Lorentzian for 3 peaks
            bounds = ([10, 2600, 15, 1275] * 3, [np.inf, 2775, 100, 1325] * 3)
            parameters, covariance = curve_fit(Lorentzian_multi_3, x_fit, y_fit, bounds=bounds)
            (amp1, cen1, wid1, y01, amp2, cen2, wid2, y02,
             amp3, cen3, wid3, y03) = parameters

            # Generate the combined fit
            x_dense = np.linspace(x_min, x_max, 1000)
            combined_y = Lorentzian_multi_3(x_dense, *parameters)

            # Find the max value of the combined curve
            max_intensity = np.max(combined_y)-y01

            # Calculate half of this max value
            half_max = max_intensity / 2 + y01

            if '_4' in file_name:
                layers=4
            elif '_5' in file_name:
                layers=5
            elif '_6' in file_name:
                layers=6

            # Find the indices where combined_y is closest to half_max
            try:
                # Get the indices of the two points closest to half_max
                indices = np.where(np.diff(np.sign(combined_y - half_max)))[0]

                if len(indices) >= 2:
                    # Extract the x-values at these indices
                    x_left = x_dense[indices[0]]
                    x_right = x_dense[indices[-1]]

                    # Calculate overall FWHM as the difference between these points
                    overall_FWHM = abs(x_right - x_left)
                    center=x_left+(overall_FWHM/2)
                    intint=integral_3(x_max, x_min, *parameters)
```

```
                    results.append((layers, file_name, max_intensity, overall_FWHM, center, intint))
                else:
                    print(f'{file_name}: Less than two half-max crossings found.')
                    overall_FWHM = np.nan

            except Exception as e:
                print(f"Error calculating FWHM for {file_name}: {e}")
                overall_FWHM = np.nan

            # Generate fit data for plotting
            fit_y = Lorentzian_multi_3(x_fit, *parameters)

        # Plot and save the fit
        print(f'{file_name}')
        output_folder = 'raman graphene/graphs'
        plt.plot(x_fit, y_fit, '.', label='data')
        plt.plot(x_fit, fit_y, 'r-', label='Lorentzian Fit')
        plt.plot(center,y01)
        plt.xlabel("Raman Shift (cm$^{-1}$)")
        plt.ylabel("Intensity (cts)")
        plt.legend(loc='upper right')
        plt.show()
        plt.close()

    except Exception as e:
        print(f"Error processing file {file_path}: {e}")

def regular_fwhm_values(params):
    return [float(fwhm) for fwhm in params['overall_FWHM']]

# Create and safe a table of file names and amplitudes
results_df = pd.DataFrame(results, columns=['layer number', 'File Name', 'Amplitude',
 'FWHM', 'center', 'integrated intensity'])
results_df.to_csv('results_table.csv', index=False)
```

Appendix C

Python Script for the Height Determination Using Two Gaussian Peaks

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit
import os

# Define Gaussian function
def Gaussian(x, amp1, mu1, sigma1, y0, amp2, mu2, sigma2):
    return (y0 + amp1 * np.exp(-((x - mu1)**2) / (2 * sigma1**2)) +
            amp2 * np.exp(-((x - mu2)**2) / (2 * sigma2**2)))

# List of files
file_list = [...]

for file_path in file_list:
    # Extract filename
    file_name = os.path.splitext(os.path.basename(file_path))[0]

    # Load data
    data = np.loadtxt(f'{file_name}.txt', delimiter=None, encoding='utf-8')
    x_data, y_data = data[:, 0] * 1e9, data[:, 1] * 1e-9  # Scale x and y data
    x_min, x_max = min(x_data), max(x_data)

    # Fit parameters
    bounds = ([0, -1.5, 0, min(y_data), 0, -0.5, 0], [2, 1, 2, 0.1, 2.5, 3, 2])

    try:
        # Fit Gaussian
        parameters, covariance = curve_fit(Gaussian, x_data, y_data, bounds=bounds)
        fit_amp1, fit_mu1, fit_sigma1, fit_y0, fit_amp2, fit_mu2, fit_sigma2 = parameters

        # Calculate height difference and store in appropriate list
        height_diff = round(abs(fit_mu2 - fit_mu1), 3)

        for force in heights.keys():
            if force in file_path:
                heights[force].append(height_diff)
```

```
        # Plot data and fit
        fit_y = Gaussian(x_data, *parameters)
        plt.plot(x_data, y_data, '.', label='Data')
        plt.plot(x_data, fit_y, 'r-', label=f'Gaussian Fit\nHeight: {height_diff} nm')
        plt.ylabel("Frequency (a.u.)")
        plt.xlabel("Height (nm)")
        plt.legend(loc='upper right')
        plt.title(f'Height distribution for {file_name}')
        plt.show()
    except Exception as e:
        print(f"Error fitting data for {file_path}: {e}")
```

Disclaimer of AI Usage

In this work, AI (`https://chatgpt.com` and `https://www.deepl.com/`) was used exclusively to correct spelling and grammar errors, as well as to refine the writing style and python scripts.

Bibliography

[1] S. V. S. Prasad, R. K. Mishra, S. Gupta, S. B. Prasad, and S. Singh, *Introduction, History, and Origin of Two Dimensional (2D) Materials*, pp. 1–9. Singapore: Springer Singapore, 2021.

[2] M.-Y. Li, C.-H. Chen, Y. Shi, and L.-J. Li, "Heterostructures based on two-dimensional layered materials and their potential applications," *Materials Today*, vol. 19, no. 6, pp. 322–335, 2016.

[3] T. Rimza, S. Singh, and P. Kumar, "Chapter 19 - two-dimensional nanomaterials for cancer application," in *Biosensor Based Advanced Cancer Diagnostics* (R. Khan, A. Parihar, and S. K. Sanghi, eds.), pp. 321–331, Academic Press, 2022.

[4] L. W. T. Ng, G. Hu, R. C. T. Howe, X. Zhu, Z. Yang, C. G. Jones, and T. Hasan, *Preface*, pp. v–vi. Cham: Springer International Publishing, 2019.

[5] Z. Guo, H. Zhang, S. Lu, Z. Wang, S. Tang, J. Shao, Z. Sun, H. Xie, H. Wang, X.-F. Yu, and P. K. Chu, "From black phosphorus to phosphorene: Basic solvent exfoliation, evolution of raman scattering, and applications to ultrafast photonics," *Advanced Functional Materials*, vol. 25, no. 45, pp. 6996–7002, 2015.

[6] K. F. Mak, C. Lee, J. Hone, J. Shan, and T. F. Heinz, "Atomically thin mos$_2$: A new direct-gap semiconductor," *Phys. Rev. Lett.*, vol. 105, p. 136805, Sep 2010.

[7] K. F. Mak, M. Y. Sfeir, J. A. Misewich, and T. F. Heinz, "The evolution of electronic structure in few-layer graphene revealed by optical spectroscopy," *Proceedings of the National Academy of Sciences*, vol. 107, no. 34, pp. 14999–15004, 2010.

[8] Z. H. Ni, H. M. Wang, J. Kasim, H. M. Fan, T. Yu, Y. H. Wu, Y. P. Feng, and Z. X. Shen, "Graphene thickness determination using reflection and contrast spectroscopy," *Nano Letters*, vol. 7, no. 9, pp. 2758–2763, 2007. PMID: 17655269.

[9] P. Blake, E. W. Hill, A. H. Castro Neto, K. S. Novoselov, D. Jiang, R. Yang, T. J. Booth, and A. K. Geim, "Making graphene visible," *Applied Physics Letters*, vol. 91, p. 063124, 08 2007.

[10] D. Bing, Y. Wang, J. Bai, R. Du, G. Wu, and L. Liu, "Optical contrast for identifying the thickness of two-dimensional materials," *Optics Communications*, vol. 406, pp. 128–138, 2018. Optoelectronics and Photonics Based on Two-dimensional Materials.

[11] Y. Hao, Y. Wang, L. Wang, Z. Ni, Z. Wang, R. Wang, C. K. Koo, Z. Shen, and J. T. L. Thong, "Probing layer number and stacking order of few-layer graphene by raman spectroscopy," *Small*, vol. 6, no. 2, pp. 195–200, 2010.

[12] L. Malard, M. Pimenta, G. Dresselhaus, and M. Dresselhaus, "Raman spectroscopy in graphene," *Physics Reports*, vol. 473, no. 5, pp. 51–87, 2009.

[13] P. Hajiyev, C. Cong, C. Qiu, and T. Yu, "Contrast and raman spectroscopy study of single- and few-layered charge density wave material: 2h-tase2," *Scientific Reports*, vol. 3, September 2013.

[14] Y. Yao, L. Ren, S. Gao, and S. Li, "Histogram method for reliable thickness measurements of graphene films using atomic force microscopy (afm)," *Journal of Materials Science & Technology*, vol. 33, no. 8, pp. 815–820, 2017.

[15] Y. Yao, L. Ren, S. Gao, and S. Li, "Histogram method for reliable thickness measurements of graphene films using atomic force microscopy (afm)," *Journal of Materials Science & Technology*, vol. 33, no. 8, pp. 815–820, 2017.

[16] R. V. Gorbachev, I. Riaz, R. R. Nair, R. Jalil, L. Britnell, B. D. Belle, E. W. Hill, K. S. Novoselov, K. Watanabe, T. Taniguchi, A. K. Geim, and P. Blake, "Hunting for monolayer boron nitride: Optical and raman signatures," *Small*, vol. 7, no. 4, pp. 465–468, 2011.

[17] A. C. Ferrari and J. Robertson, "Raman spectroscopy of amorphous, nanostructured, diamond–like carbon, and nanodiamond," *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, vol. 362, pp. 2477 – 2512, 2004.

[18] C. Thomsen and S. Reich, "Double resonant raman scattering in graphite," *Phys. Rev. Lett.*, vol. 85, pp. 5214–5217, Dec 2000.

[19] M. Krečmarová, D. Andres-Penares, L. Fekete, P. Ashcheulov, A. Molina-Sánchez, R. Canet-Albiach, I. Gregora, V. Mortet, J. P. Martínez-Pastor, and

J. F. Sánchez-Royo, “Optical contrast and raman spectroscopy techniques applied to few-layer 2d hexagonal boron nitride,” *Nanomaterials*, vol. 9, no. 7, 2019.

[20] F. Moreno-Herrero and J. Gomez-Herrero, *AFM: Basic Concepts*, ch. 1, pp. 1–34. John Wiley and Sons, Ltd, 2012.

[21] A. Trache and G. A. Meininger, “Atomic force microscopy (afm),” *Current Protocols in Microbiology*, vol. 8, no. 1, pp. 2C.2.1–2C.2.17, 2008.

[22] Oxford Instruments, “Afm: Exploring tapping mode and am-fm.” `https://afm.oxinst.com/outreach/tapping-mode-for-afm-am-fm`. Last accessed 18.02.2025.

[23] Park Systems Corporation, “Tapping mode.” `https://www.parksystems.com/en/products/research-afm/AFM-modes/imaging-modes/tapping-mode`. Last accessed 09.02.2025.

[24] N. Jalili and K. Laxminarayana, “A review of atomic force microscopy imaging systems: application to molecular metrology and biological sciences,” *Mechatronics*, vol. 14, no. 8, pp. 907–945, 2004.

[25] C. J. Shearer, A. D. Slattery, A. J. Stapleton, J. G. Shapter, and C. T. Gibson, “Accurate thickness measurement of graphene,” *Nanotechnology*, vol. 27, p. 125704, feb 2016.

[26] Bruker Corporation, “Application note # 133 introduction to bruker’s scanasyst and peakforce tapping afm technology,” `https://www.epfl.ch/research/facilities/cmi/wp-content/uploads/2020/03/Introduction_to_Brukers_ScanAsyst_and_PeakForce_Tapping.pdf`.

[27] K. Xu, W. Sun, Y. Shao, F. Wei, X. Zhang, W. Wang, and P. Li, “Recent development of peakforce tapping mode atomic force microscopy and its applications on nanoscience,” *Nanotechnology Reviews*, vol. 7, no. 6, pp. 605–621, 2018.

[28] F. Schäffel, “Chapter 2 - the atomic structure of graphene and its few-layer counterparts,” in *Graphene* (J. H. Warner, F. Schäffel, A. Bachmatiuk, and M. H. Rümmeli, eds.), pp. 5–12, Elsevier, 2013.

[29] J. Anjilivelil and D. Bhattacharyya, *Graphene: an introduction*, pp. 3–6. 06 2023.

[30] J. H. Warner, F. Schäffel, A. Bachmatiuk, and M. H. Rümmeli, "Chapter 3 - properties of graphene," in *Graphene* (J. H. Warner, F. Schäffel, A. Bachmatiuk, and M. H. Rümmeli, eds.), pp. 62–64, Elsevier, 2013.

[31] A. Castro Neto, F. Guinea, N. Peres, K. Novoselov, and A. Geim, "The electronic properties of graphene," *Review of Modern Physics*, vol. 81, 10 2007.

[32] C. Leong and B. Roy, "Non-hermitian catalysis of density-wave orders on euclidean and hyperbolic lattices," 01 2025.

[33] L. Menacho, M. Carrasco, and Z. Ayala, "Comparison between monolayer and bilayer graphene energy bands using the tight binding model," *Journal of Physics: Conference Series*, vol. 1143, p. 012022, dec 2018.

[34] E. Sharin and R. Tihonov, "Two-dimensional carbon structures study within density functional theory," vol. 1907, p. 030028, 11 2017.

[35] *Appendix A: The Fresnel Laws*, pp. 365–385. John Wiley & Sons, Ltd, 2009.

[36] P. Rostron, S. Gaber, and D. Gaber, "International journal of engineering and technical research (ijetr)," 2016.

[37] C. L. Haynes, A. D. McFarland, and R. P. Van Duyne, "Surface-enhanced raman spectroscopy," *Analytical Chemistry*, vol. 77, no. 17, pp. 338 A–346 A, 2005.

[38] L. Staveley, *The Characterization of Chemical Purity: Organic Compounds*, p. 149. Butterworth-Heinemann, 2016.

[39] Q. Zhang, K. Wu, H. Qian, B. Ramachandran, and F. Jiang, "The advances of characterization and evaluation methods for the compatibility and assembly structure stability of food soft matter," *Trends in Food Science & Technology*, vol. 112, pp. 753–763, 2021.

[40] G. S. Papanai, I. Sharma, and B. K. Gupta, "Probing number of layers and quality assessment of mechanically exfoliated graphene via raman fingerprint," *Materials Today Communications*, vol. 22, p. 100795, 2020.

[41] Y. Liu, Z. Liu, W. S. Lew, and Q. Wang, "Temperature dependence of the electrical transport properties in few-layer graphene interconnects," *Nanoscale research letters*, vol. 8, p. 335, 07 2013.

[42] A. K. Ott and A. C. Ferrari, "Raman spectroscopy of graphene and related materials," in *Encyclopedia of Condensed Matter Physics (Second Edition)*

(T. Chakraborty, ed.), pp. 233–247, Oxford: Academic Press, second edition ed., 2024.

[43] E. H. Hasdeo, A. R. T. Nugraha, M. S. Dresselhaus, and R. Saito, "Fermi energy dependence of first- and second-order raman spectra in graphene: Kohn anomaly and quantum interference effect," *Phys. Rev. B*, vol. 94, p. 075104, Aug 2016.

[44] H. Wahab, S. Ali, A. Abdul Hussein, and A. Abdul, "Synthesis and characterization of graphene by raman spectroscopy citation," *Journal of Materials Sciences and Applications*, vol. 1, pp. 130–135, 06 2015.

[45] A. C. Ferrari, J. C. Meyer, V. Scardaci, C. Casiraghi, M. Lazzeri, F. Mauri, S. Piscanec, D. Jiang, K. S. Novoselov, S. Roth, and A. K. Geim, "Raman spectrum of graphene and graphene layers," *Phys. Rev. Lett.*, vol. 97, p. 187401, Oct 2006.

[46] A. C. Ferrari, J. C. Meyer, V. Scardaci, C. Casiraghi, M. Lazzeri, F. Mauri, S. Piscanec, D. Jiang, K. S. Novoselov, S. Roth, and A. K. Geim, "Raman spectrum of graphene and graphene layers," *Phys. Rev. Lett.*, vol. 97, p. 187401, Oct 2006.

[47] D. Graf, F. Molitor, K. Ensslin, C. Stampfer, A. Jungen, C. Hierold, and L. Wirtz, "Spatially resolved raman spectroscopy of single- and few-layer graphene," *Nano Letters*, vol. 7, no. 2, pp. 238–242, 2007. PMID: 17297984.

[48] B. Lautrup, *Physics of Continuous Matter: Exotic and everyday phenomena in the macroscopic world*, pp. 125–128. 04 2004.

[49] NanoAndMore USA Corp., "More about afm cantilevers." `https://www.nanoandmore.com/more-about-afm-cantilevers`. Last accessed 07.02.2025.

[50] J. L. Hutter and J. Bechhoefer, "Calibration of atomic-force microscope tips," *Review of Scientific Instruments*, vol. 64, pp. 1868–1873, 07 1993.

[51] T. A. Efimov, E. A. Rassolov, B. G. Andryukov, T. S. Zaporozhets, and R. V. Romashko, "Calculation of resonant frequencies of silicon afm cantilevers," *Journal of Physics: Conference Series*, vol. 1439, p. 012006, jan 2020.

[52] C. T. Gibson, G. S. Watson, and S. Myhra, "Determination of the spring constants of probes for force microscopy/spectroscopy," *Nanotechnology*, vol. 7, p. 259, sep 1996.

[53] R. Reifenberger, *Fundamentals Of Atomic Force Microscopy - Part I: Foundations*, ch. 1.4, pp. 10–15. Lessons From Nanoscience: A Lecture Notes Series, World Scientific Publishing Company, 2015.

[54] P. Schwerdtfeger and D. J. Wales, "100 years of the lennard-jones potential," *Journal of Chemical Theory and Computation*, vol. 20, no. 9, pp. 3379–3405, 2024. PMID: 38669689.

[55] R. K. Puddy, P. H. Scard, D. Tyndall, M. R. Connolly, C. G. Smith, G. A. C. Jones, A. Lombardo, A. C. Ferrari, and M. R. Buitelaar, "Atomic force microscope nanolithography of graphene: Cuts, pseudocuts, and tip current measurements," *Applied Physics Letters*, vol. 98, Mar. 2011.

[56] Park Systems Corporation, "Contact mode is a standard imaging mode of atomic force microscopy (afm) that can obtain topographic information on a wide range of sample types.." `https://www.parksystems.com/en/products/research-afm/AFM-modes/imaging-modes/ContactMode`. Last accessed 11.02.2025.

[57] F. Serry and Bruker Corporation, "Improving the accuracy of afm force measurements: The thermal tune solution to the cantilever spring constant problem." `https://www.bruker-nano.jp/library/57e489500c201abd043451c7/57fde5d24e5256d4724aeb91.pdf`. Last accessed 03.01.2025.

[58] mikromasch, "Hq:nsc18/al bs." `https://www.spmtips.com/afm-tip-hq-nsc18-al-bs`. Last accessed 13.02.2025.

[59] S. Belikov, J. Alexander, C. Wall, I. Yermolenko, S. Magonov, and I. Malovichko, "Thermal tune method for afm oscillatory resonant imaging in air and liquid," in *2014 American Control Conference*, pp. 1009–1014, 2014.

[60] Bruker Corporation, Santa Barbara, CA, *Measuring Thermal Noise in Dimension Icon User Guide*, 2010. Available at `https://www.nanophys.kth.se/nanolab/afm/icon/bruker-help/Content/Probe%20and%20Sample%20Guide/ThermalTune/MeasuringThermalNoise.htm`.

[61] Bruker Corporation, Santa Barbara, CA, *Determine Cantilever Spring Constant by Thermal Tune in Dimension Icon User Guide*, 2010. Available at `https://www.nanophys.kth.se/nanolab/afm/icon/bruker-help/Content/Probe%20and%20Sample%20Guide/ThermalTune/ThermalTune.htm#kanchor80`.

[62] Y. Huang, E. Sutter, N. N. Shi, J. Zheng, T. Yang, D. Englund, H.-J. Gao, and P. Sutter, "Reliable exfoliation of large-area high-quality flakes of graphene and other two-dimensional materials," *ACS Nano*, vol. 9, no. 11, pp. 10612–10620, 2015. PMID: 26336975.

[63] BudgetSensors, "Tap300al-g." `https://www.budgetsensors.com/tapping-mode-afm-probe-aluminum-tap300al`. Last accessed 13.02.2025.

[64] D. Nečas and P. Klapetek, "Gwyddion: An open-source software for spm data analysis," *Central European Journal of Physics*, vol. 10, 12.

[65] P. Klapetek, D. Nečas, and C. Anderson, "Gwyddion user guide." `http://gwyddion.net/download/user-guide/gwyddion-user-guide-en.pdf`, 2004. Last accessed 26.02.2025.

[66] Bruker Corporation, Santa Barbara, CA, *Prepare and Load the Probe Holder in Dimension Icon User Guide*, 2010. Available at `https://www.nanophys.kth.se/nanolab/afm/icon/bruker-help/Content/Basic%20AFM%20Operation/Prepare%20and%20Load%20the%20Cantilever%20Holder.htm`.

[67] Bruker Corporation, Santa Barbara, CA, *Calibrate the Spring Constant Using Thermal Tuning in Dimension Icon User Guide*, 2010. Available at `https://www.nanophys.kth.se/nanolab/afm/icon/bruker-help/Content/Basic%20AFM%20Operation/Prepare%20and%20Load%20the%20Cantilever%20Holder.htm`.

I hereby declare that this thesis is my own work and I have not used any sources and aids other than those stated in the thesis.

Hiermit erkläre ich, die vorliegende Arbeit selbständig verfasst zu haben und keine anderen als die in der Arbeit angegebenen Quellen und Hilfsmittel benutzt zu haben.

Ylvi Haizmann
Ludwig-Maximilians-Universität München

München, 07 March, 2025