

Height profile analysis of Bi islands deposited on Epitaxial Graphene using Linear Constrained Optimization Techniques

Chitran Ghosal

1 Abstract

Bi deposited (in Ultra High Vacuum) on HOPG or other graphitic substrates does not form a wetting layer, but decomposes into several tiny crystalline islands. It would be expected from bland geometric arguments, that graphite being hexagonal would host Bi crystallites of a hexagonal nature, for ex. of a Bi(111) crystallinity, but interestingly that turns out not to be the case. Bi forms extremely tiny Bi(110) crystallites on top of the HOPG substrate [1]. This behaviour implies that simple geometric arguments can't be used to explain the self assembly behaviour and some other energetic considerations need to be taken into the picture. It is also well known that on HOPG, Bi prefers to arrange itself in islands with an odd no. of monolayers (3ML, 5ML....) [1]. The explanation of this bilayer-like model is difficult to conclude through experimental work alone and theoretical models have been proposed in literature to deal with this behaviour, the most accepted one being the relaxation of two layers of Bi(110) into a joint bilayer like structure termed the *black phosphorous*(BP) phase [2]. However, this behaviour needs to be tested on other substrates to be able to benchmark the theory for precision. The formation of these BP structures should dampen interaction between subsequent bilayers, however this has also been shown not to be the case [1]. Hence, adding more case-studies of Bi/X substrates to the existing database would further enhance the theory behind the self-assembly processes.

2 Introduction

Bi is a semimetal with many interesting properties. It has been shown that Pb deposited on chemically inert substrates, like HOPG, tend to arrange themselves in islands which give rise to magic heights or rather a statistical preference of some heights over others [3], [4]. The semimetal Bi, being in close proximity to Pb in the periodic table could be expected to relay characteristics and growth morphologies paralleling that of Pb. The magic heights of Pb islands were closely correlated to the Fermi wavelengths, λ_F , of the materials in question[5], [6]. However, substrate/interface effects also need to be considered in order to model the heurestics of these QWS(Quantum Well States) [5]. In order to hollistically model the entirety of Quantum Size Effects(QSE) as a function of the various factors like size-confinement, fermi-wavelength, heavy-fermionic effects and substrate interface effects, a statistically accurate study of the height distribution of the deposited Bi needs to be conducted at different coverages and different substrates.

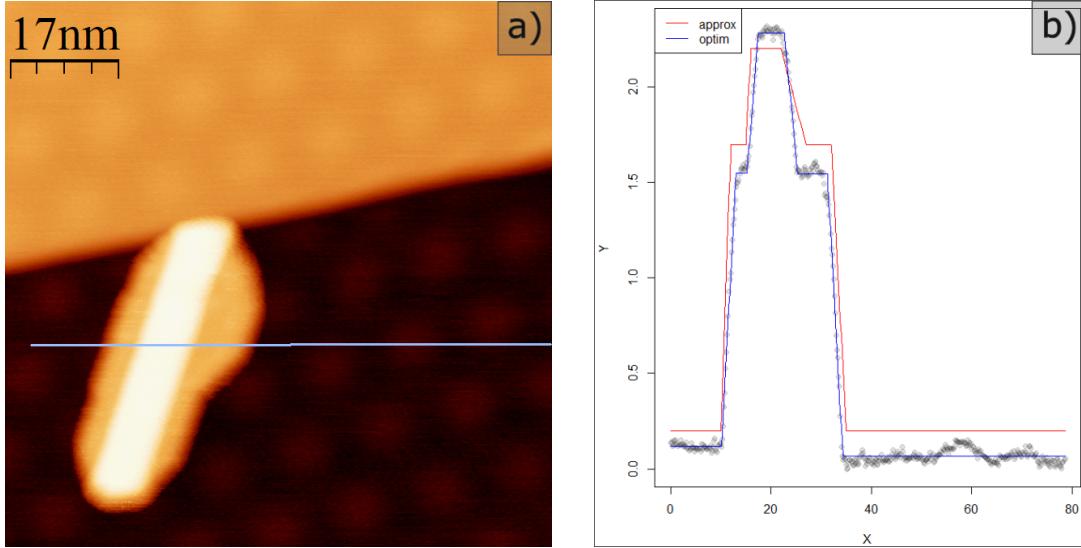


Figure 1: (a) STM image of a Bi island on HOPG, Image extracted at settings of 100mV, 500pA. (b) Line scan across fig.1a plotted and fit to the code in question. All units plotted in fig.1b are in nm.

Height distribution profiles for Bi/HOPG are already available in literature [1], but have been repeated in this study as a measure of reproducibility of the literature, while height distributions for Bi on epitaxial graphene(EG) were extracted for study and comparison. EG was chosen as a substrate because the band structures of HOPG and EG should differ mostly as a result of substrate doping and should not suffer from renormalization scenarios[7].

In order to realize an accurate experimental study of the height profiles, this work attempts to write and formulate a non-linear optimization model which minimizes a squared regression loss function, which is the objective function for the stated optimization scenario. Iterative and consecutive use of this model fits line profiles to island heights, as will be shown, and builds a sampling distribution which is indicative of the population present in the large scale STM images of the underlying study. Thus it allows us to construct a *predictive* height distribution of the underlying population of Bi/EG islands.

These height distributions are then correlated with the electronic structure of these island heights which are studied using Scanning Tunneling Spectroscopy(STS). The STS spectra scale as Density of States(DOS) as predicted and by the Tersoff-Hamann models[8]. Hence, variations of island heights have also been electronically studied to further understand the experimental distributions of island heights.

As a final proof of concept, the optimization model is judged on the basis of increasing gaussian noise added to a simulated data-set and the model performance is evaluated based on the spread of regression coefficients, which in this case is the global minima of the objective function in parameter/coefficient space. Also the absolute values of the loss-function at their minima is plotted against the increasing spread of the gaussian noise variance in order to judge the robustness of the model over the entire spread of the noise, which gives a cue as to whether an upper limit is reached beyond which the model collapses with decreasing SNR.

3 The Optimization Problem

3.1 Defining the Parameter Space

In order to introduce the formulation of the problem, a single line-profile is shown in fig.1b for clarity. The line profile is extracted from fig.1a. The red profile marked approx in the legend in fig.1b is the initial guess, denoted as vector \vec{v}_θ henceforth, while the blue profile marked optim is the result of the minimization of the objective function.

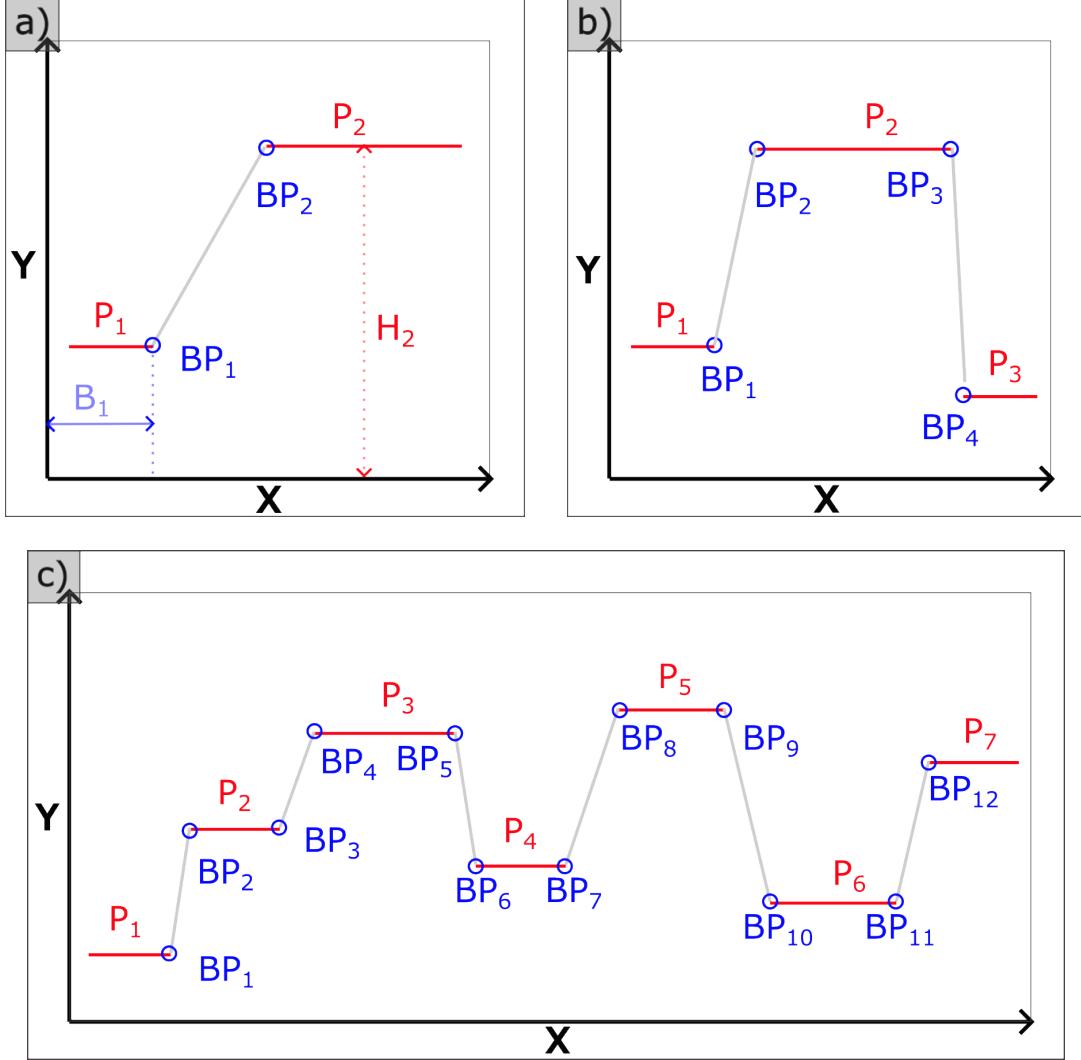


Figure 2: (a) A two breakpoint, two plateau profile. (b) A four breakpoint, three plateau profile. (c) A twelve breakpoint, seven plateau profile.

This objective function is a squared loss function which is denoted as $\mathcal{L}(\vec{v}, \vec{X}, \vec{Y})$ henceforth. The function is minimized w.r.t. \vec{v} with an initial starting value of \vec{v}_θ . In the process of deriving $\mathcal{L}(\vec{v}, \vec{X}, \vec{Y})$, an inspection of the profiles shown in fig.2 is needed. As can be seen, the

line-profiles comprise of three physical parts, marked in red, gray and blue and named *plateau*, *slope* and *breakpoints* respectively. Since the *breakpoints* are the corners where the *plateau* and the *slope* intersect, consequently defining any two automatically leads to the definition of the third. As a result, the *plateau* and the *breakpoint* have been chosen as the elements of interest for this work. Each *plateau* and *breakpoint* is denoted in fig.2 as P_i and BP_j respectively, such that $i, j \in \mathbb{Z}^+$

As a consequence of the aforementioned formulation, collections of the sequence of relevant elements in fig.2a, fig.2b and fig.2c are placed in sets \mathcal{S}_a , \mathcal{S}_b and \mathcal{S}_c as shown in eq.1, eq.2 and eq.3 respectively.

$$\mathcal{S}_a = \{P_1, BP_1, BP_2, P_2\} \quad (1)$$

$$\mathcal{S}_b = \{P_1, BP_1, BP_2, P_2, BP_3, BP_4, P_3\} \quad (2)$$

$$\begin{aligned} \mathcal{S}_c = & \{P_1, BP_1, BP_2, P_2, BP_3, BP_4, P_3, BP_5, BP_6, P_4, BP_7, BP_8, P_5, BP_9, BP_{10}, P_6, \\ & BP_{11}, BP_{12}, P_7\} \end{aligned} \quad (3)$$

As can be seen from fig.2 and eq.1 through eq.3, sets \mathcal{S}_k offer certain system invariants shown in Invariants1.

1. Each \mathcal{S}_k begins and ends in a P_i
2. Each P_i , if not the first or last element, is preceded and succeeded with a BP_j
3. BP_j s always occur in pairs

Invariants 1: Set of basic invariants

These invariants enforce certain structures onto \mathcal{S}_k . As a result, the no. of BP_j s are always even. Henceforth, $n = |BP_j|$, represents the cardinality of BP_j , $p = |P_i|$, represents the cardinality of P_i s and $L = n + p = |\mathcal{S}_k|$ represents the cardinality of the entire set.

$$n = 2m \mid m \in \mathbb{Z}^+ \quad (4)$$

$$p = \frac{n}{2} + 1 \quad (5)$$

$$L = 3\frac{n}{2} + 1 \quad (6)$$

The equations eq.4 through eq.6 are a consequence of the invariants presented in Invariants1. Additionally, the elements in \mathcal{S}_k are simply objects and do not form a closed group under any operation. Thus a mapping of the form $f \mid f : \mathcal{S}_k \rightarrow \mathbb{R}^L$ was defined w.r.t the heights of P_i and x-coordinates of BP_j as shown in fig.2a and formulated in eq.7.

$$f(x) = \begin{cases} H_i, & \text{if } x \text{ is a } P_i \\ B_j, & \text{if } x \text{ is a } BP_j \end{cases} \quad (7)$$

At this point, the output of $f(x)$ in eq.7 can be redefined as the previously defined vector \vec{v} , which forms the parameter space of the function $\mathcal{L}(\vec{v}, \vec{X}, \vec{Y})$. Hence, it can be easily retorted that $f : \mathcal{S}_k \rightarrow \vec{v} \mid \vec{v} \in \mathbb{R}^L$.

For convenience the terms in \vec{v} were rearranged in eq.8

$$\vec{v} = \{B_1, B_2, B_3, \dots, B_n, H_1, H_2, H_3, \dots, H_p\} \quad (8)$$

$$= \{v_1, v_2, v_3, \dots, v_n, v_{n+1}, v_{n+2}, v_{n+3}, \dots, v_{n+p}\} \quad (9)$$

Hence, a relationship can be drawn between the indexes of v_k and those of H_i and B_j as shown in eq.10

$$v_k = \begin{cases} B_k, & \text{if } k \leq n \\ H_{k-n}, & \text{if } k > n \end{cases} \quad (10)$$

3.2 The Regression Model

Using the parameter space \vec{v} definition in eq.8 through eq.10, a regression model is defined with the initial estimates \vec{v}_θ of \vec{v} . The estimate \vec{v}_θ is designated with members ${}^k v_\theta$ as shown in eq.11 and eq.12.

$$\vec{v}_\theta = \{{}^1 v_\theta, {}^2 v_\theta, {}^3 v_\theta, \dots, {}^L v_\theta\} \quad (11)$$

$$= \{{}^k v_\theta \mid k \in [1, L] \cap \mathbb{Z}\} \quad (12)$$

The approx profile (denoted in red in fig.1b) was arrived at using the function $g(x, \vec{v}_\theta) \mid x \in \mathbb{R} \mid \vec{v}_\theta \in \mathbb{R}^L$. Thus the function $g : \mathbb{R}^{L+1} \rightarrow \mathbb{R}$ is defined in eq.13.

$$g(x, \vec{v}_\theta) = \begin{cases} {}^{n+1} v_\theta, & x \leq {}^1 v_\theta \\ {}^{n+1+k/2} v_\theta, & {}^k v_\theta \leq x \leq {}^{k+1} v_\theta, k = 2m, k \leq n-1 \\ \left[\frac{{}^{n+(k+3)/2} v_\theta - {}^{n+(k+1)/2} v_\theta}{{}^{k+1} v_\theta - {}^k v_\theta} \right] (x - {}^k v_\theta) + {}^{n+\frac{k+1}{2}} v_\theta, & {}^k v_\theta \leq x \leq {}^{k+1} v_\theta, k = 2m \pm 1, k \leq n-1 \\ {}^{n+p} v_\theta, & x \geq {}^n v_\theta \end{cases} \quad (13)$$

The four conditions seen in eq.13 are

1. when $x \leq B_1 = {}^1 v_\theta \implies y = H_1 = {}^{n+1} v_\theta$
2. when x lies between an even and a odd j in BP_j , $y = H_{1+k/2}$
3. when x lies between an odd and even j in BP_j , then we hit a *slope*
4. when $x \geq BP_n = {}^n v_\theta \implies y = H_p = {}^{n+p} v_\theta$

Invariants 2: Derived set of invariants

Having derived eq.13, it must be stated that since \vec{v}_θ is a constant for the parameter space w.r.t the loss $\mathcal{L}(\vec{v}, \vec{X}, \vec{Y})$, we need a function $f(x, \vec{v})$ using the variable parameter \vec{v} which can be iterated over by the optimization engine. Paralleling eq.13, we define eq.14 using the function $f \mid f : \mathbb{R}^{L+1} \rightarrow \mathbb{R}$.

$$f(x, \vec{v}) = \begin{cases} v_{n+1}, & x \leq v_1 \\ v_{n+1+k/2}, & v_k \leq x \leq v_{k+1}, k = 2m, k \leq n-1 \\ \left[\frac{{}^{n+(k+3)/2} v_k - {}^{n+(k+1)/2} v_k}{v_{k+1} - v_k} \right] (x - v_k) + v_{n+\frac{k+1}{2}}, & v_k \leq x \leq v_{k+1}, k = 2m \pm 1, k \leq n-1 \\ v_{n+p}, & x \geq v_n \end{cases} \quad (14)$$

Note that $g(x, \vec{v}_\theta)$ is derived within lines 41 and 64 of the user defined code in R, using the *prof_struct* function. See sec.7, listing.1.

3.3 The Objective function

Having defined $f(x, \vec{v})$, we move to define our objective function. Let $h(\vec{v})$ be our objective function and let $x_i, y_i, i \in [1, N] \cap \mathbb{Z}$ be our x-y dataset. Then, we could define $h(\vec{v})$ as shown in eq.15.

$$h(\vec{v}) = \sum_{i=1}^n (y_i - f(x_i, \vec{v}))^2 \quad (15)$$

Note that the minimization of the function $h(\vec{v})$ would minimize the SSE for the function $f(x, \vec{v})$ w.r.t the x-y dataset. In light of proper notation, we can now define our loss function, $\mathcal{L}(\vec{v}, \vec{X}, \vec{Y}) = h(\vec{v})$.

As far as the constraints are concerned, we would want $B_{i+1} \geq B_i$ since this would ensure all *ascending slopes* have a positive gradient and ensure that all *descending slopes* have a negative gradient. Also we would want that the optimized heights should lie between the extremes of our dataset, i.e. $\min(\vec{Y}) \leq H_k \leq \max(\vec{Y})$. Hence, formalizing these statements, we get the constraints in eq.16 and eq.17.

$$v_{k+1} - v_k \geq 0 \quad \forall k \leq n-1 \mid k \in \mathbb{Z}^+ \quad (16)$$

$$\min(\vec{Y}) \leq v_k \leq \max(\vec{Y}) \quad \forall k \geq n+1 \mid k \in \mathbb{Z}^+ \quad (17)$$

Hence, the final optimization problem can be written as eq.18

$$\begin{aligned} & \underset{\vec{v}}{\text{minimize}} && \mathcal{L}(\vec{v}, \vec{X}, \vec{Y}) \\ & \text{subject to:} && v_{k+1} - v_k \geq 0 \quad \forall k \leq n-1 \mid k \in \mathbb{Z}^+ \\ & && \min(\vec{Y}) \leq v_k \leq \max(\vec{Y}) \quad \forall k \geq n+1 \mid k \in \mathbb{Z}^+ \end{aligned} \quad (18)$$

It shoul be noted from eq.15 that the objective is not linear in \vec{v} , but the constarints are linear (see eq.16 and eq.17) inequalities w.r.t \vec{v} .

4 Results

In this section we discuss the results achieved using the characterization algorithm mentioned in the previous section.

As a first attempt, 6H-SiC was used wherein the epitaxial graphene(EG henceforth) on the samples were grown using the PASG method [9]. These samples were then annealed in an ultra high vacuum preparation chamber with base pressures of 10^{-10} mbar to temperatures of 800°C . The pressure in the chamber rose to orders of around 5.5×10^{-9} mbar indicating the desorption of species from the samples. In a final attempt to characterize the samples, room temperature STM was conducted on the samples wherein the atomic structure of epitaxial graphene was succesfully imaged in an analysis chamber with pressures in the 5×10^{-11} mbar regime. The atomically resolved images of graphene can be seen (inset) in fig.3a, 3b and fig.3c and the 2D-fourier transform of the image was used to specify the directions shown in fig.3. Bi was then deposited from the preparation chamber (base pressures of 10^{-10} mbar) using a home built evaporator with a ceramic knudsen cell attached with a quartz microbalance used to read off the flux of the epitaxy process. The varied flux for the micrographs of fig.3a, 3b and 3c are 4ML, 6ML and 40ML as shown in the images. The deposition pressures for the Bi/EG was maintained at 10^{-8} mbar. The STM tips used for the imaging were freshly cut

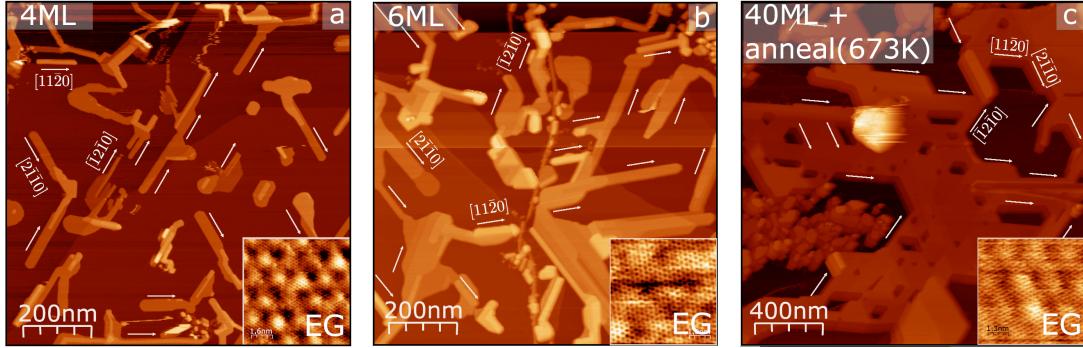


Figure 3: (a) STM image of Bi island on HOPG at 4ML of deposited Bi flux, Image extracted at settings of 100mV, 200pA. (b) STM image of Bi island on HOPG at 6ML of deposited Bi flux, Image extracted at settings of 100mV, 200pA. (c) STM image of Bi island on HOPG at 40ML of deposited Bi flux followed by a post annealing session for 30 min at temperatures of 673 K, Image extracted at settings of 100mV, 200pA.

Pt-Ir tips from a Pt-Ir wire of 0.25 mm diameter.

Directly visible in fig.3a and 3b, the Bi forms needle like structures with preferred directions being the $\langle 11\bar{2}0 \rangle$ planes w.r.t graphene, which is the armchair direction of graphene. This tri-directional extension of growth makes sense because graphene belongs to the D_{6h} point group and its symmetry would afford this growth geometry. However this behaviour has also been seen with Bi/HOPG growth and counterintuitively the atomic structure of the Bi on HOPG has been shown not to be of a hexagonal Bi(111) variant, rather a squarish lattice of the Bi(110) type [1]. These measurements were repeated for the Bi/EG system as shown

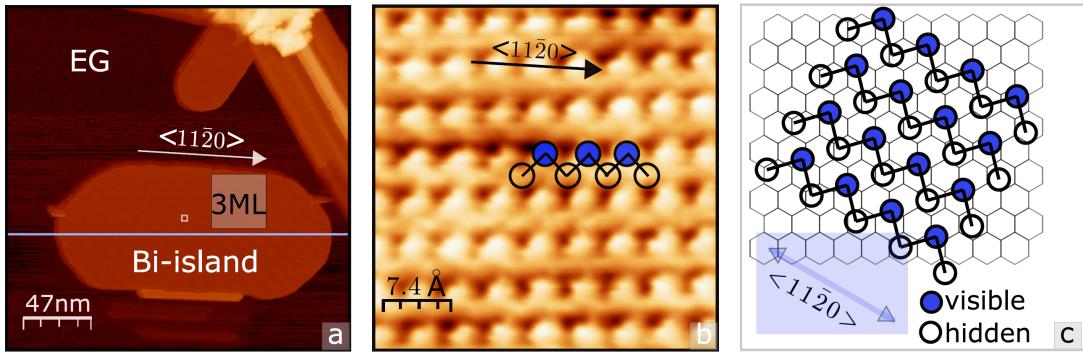


Figure 4: (a) STM image of a 3ML high Bi island on EG. Image settings: 100mV, 200pA. (b) Atomically resolved STM image of the Bi island shown in (a). Image settings: 100mV, 200pA. (c) 2D-ball and stick model of the atomic lattice imaged in (b), combined with growth direction along the armchair direction of graphene.

in fig.4. As can be seen in fig.4a, the image shows a 3ML (Height information was extracted using the algorithm etched in the *The Optimization Problem* section) high Bi island growing along the $\langle 11\bar{2}0 \rangle$ direction as marked. A small, empty square is marked on the island (in

white). Fig.4b shows a zoom into this square wherein the rectangular Bi(110) lattice for this island is visible. Consequently all islands visible on the surface were confirmed to be of the Bi(110) nature. The lattice constants of the Bi(110) were verified (using 2D-fourier transforms) to be within reasonable approximations of the bulk Bi(110) structure. This (110) like structure, as mentioned before, has also been reported for Bi/HOPG[1] and for Bi/Si(111)[2]. Since Si(111) belongs to the symmetry group C_{3v} and HOPG belongs to the symmetry group D_{3d} , which are both hexagonal in nature, but Bi(110) has a C_{2h} symmetry with a two-fold rotational symmetry axis perpendicular to the surface, naive geometric arguments leading to the structural commensurability of the substrate and overlayer lattice cannot be used in this case. Hence, there need to be some energetic considerations which do not scale with structural geometry which give rise to the said Bi(110) structures. This incommensurability of lattice geometries compounded with the fact that the Bi, as can be seen from fig.3a and fig.3b, does not wet the graphene as much as it likes to grow in height, directly makes a case for Bi having much larger cohesive forces within itself rather than having adhesion to the substrate and its geometry. In fact, the growth of Bi(110) islands on the Si(111), rather than an adhesive Bi(111), has been modelled exactly using an interplay of the cohesive/adhesive energies with increasing ML of the island stacks[2]. A further foray into this argumentation can be made on the basis of Bi(111)/Si(111) grown by holding the substrate to temperatures as low as 200 K[10] instead of holding at room temperature, as is the case with this work. It suffices to understand that the sticking probability of a ballistically travelling molecule in vacuum would increase with decreasing substrate temperature thus increasing the adhesion between the overlayer and substrate.

Nevertheless, having drawn two analogues, namely armchair directed growth($\langle 1\bar{1}20 \rangle$) and cohesive structural behaviour(Bi(110)), the third important factor of the Bi/HOPG was tested for Bi/EG, which was the existence of preferential heights, reported as odd no of MLs were seen to be more prevalent on HOPG[1] instead of it being uniformly distributed.

In order to realize the ML distribution, height profiles of the islands were extracted (see for ex. fig.1a) and fit to the function $f(x, \vec{v})$ as shown in fig.1b. Then the average substrate height was set to zero and the height of the island was calculated. The island height was then divided by 0.33 nm which is the unit cell height of Bi(110) in the bulk [2]. Then the result of this division is rounded up or rounded down depending on its closest integer number¹. Followed by the characterisation of each island into an integer ML number, the frequency of the existing no. of islands against each ML height was mapped to a histogram depending on the deposition coverage onto the substrate, which was read by the flux reading of the ceramic microbalance. This was plotted in fig.5. From fig.5a through fig.5c, the flux was plotted at the top right hand corner of the image and it can directly be seen that as the deposition coverage is doubled and then tripled, the substrate steps of EG is still visible in fig.3a and fig.3b, while the height profile preferences in fig.5 shift to higher ML values, indicating an inability to wet the substrate, hence feeding back into the cohesive coagulation of Bi versus adhesive wetting. What is of even higher interest is that instead of the preference for an odd no. of ML heights seen with Bi/HOPG[1], Bi/EG shows a preference for an even no. of ML heights as seen in fig.5. Also of interest is the fact that the 4ML structure retains the spot for the most probable island height through coverages of 2ML and 4ML (see fig.5a and fig.5b). This thermodynamic stability of the 4ML structure might be related to certain energetically favourable criterion which has been investigated in the verification section. Thus the his-

¹This method makes the sweeping generalization that the thin film vertical lattice constant stays the same as the bulk vertical lattice constant. Hence, no real second order quantum effects have been considered.

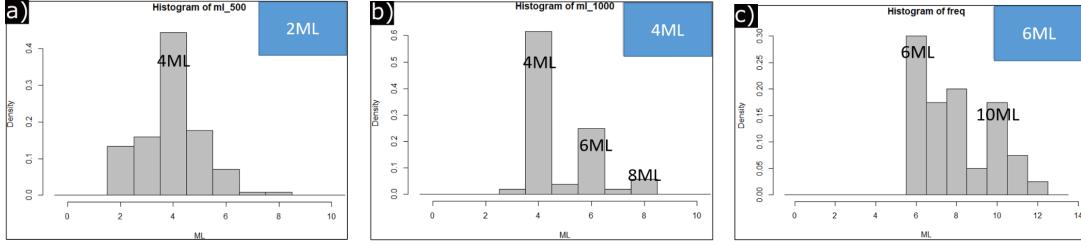


Figure 5: (a) Histogram of island heights of Bi(110) islands on EG, deposited for a Bi flux of 2ML. (b) Histogram of island heights of Bi(110) islands on EG, deposited for a Bi flux of 4ML. (c) Histogram of island heights of Bi(110) islands on EG, deposited for a Bi flux of 6ML.

tograms of fig.5 clearly prove that unlike the preference of the HOPG substrate for Bi(110) island heights of an odd no. of monolayers[1], the EG substrate has a selective preference for an even no. of monolayers. Attributing causality to this behaviour is not an easy task and at this point theoretical modelling would be the preferred approach because of a lack of any well understood experimentally verifiable parameter/attribute which could be tweaked to further design experiments in pursuit of this question. However in such cases, it is not unusual to assign correlations as causations with caution. A trivial attempt at doing so leads us to the structure of HOPG in clash against the structure of EG. EG is essentially a single layer of graphene with an insulating/semiconducting SiC substrate below, whereas HOPG is several layers of graphene or essentially, laboratory grade graphite. This difference could cause a difference in the way the charge cloud/distribution of the bottom-most layer of the Bi(110) islands gets screened/imaged by the substrate as a result of the different vertical dielectric behaviour of the EG, compared to that of HOPG. However, this line of questioning, is at the moment, best described as a hypothesis to be studied using a different set of experimental design.

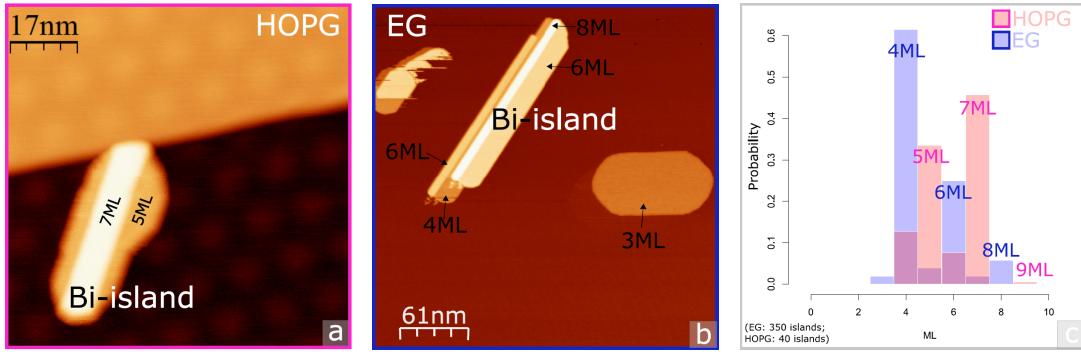


Figure 6: (a) STM image of Bi/HOPG showing preference for odd no. of monolayers. Image settings: 100mV, 200pA. (b) STM image of Bi/EG showing preference for even no. of monolayers. Image settings: 100mV, 200pA. (c) Histogram of island heights of Bi(110) islands on EG/HOPG showing the even-odd discrepancy on the same histogram.

In order to bat for completeness, in spite of literature being available, Bi was also deposited

on HOPG for this work and their height profiles studied using the exact same analytical technique which was used to study Bi/EG. As can be seen in fig.6a, the HOPG substrate shows the existence of a 7ML island growing on a 5ML base, whereas in the case of fig.6b, the EG substrate shows the whole range of available island heights, from 4ML through 8ML growing mostly on the (presumably)thermodynamically stable 4ML base. Interestingly, also seen in the image is a 3ML island, but 2ML islands could never be found using STM when specifically being searched for. The existence of the 2ML with a probability density between 0.1 and 0.2 in fig.5a is attributed to measurement noise causing erroneous reading of the 3ML height which forced the ML integer to overflow into the previous histogram bin. The lack of a 2ML or 1ML structure also could not be experimentally explained and needs some theoretical modelling probably in the direction of energy of structure formation.

5 Verification

In order to verify the preference of the population distribution towards the 4ML + 2nML, the *Verification* section, sec.5, is divided into two subsections, namely the *Scanning Tunneling Spectroscopy*(STS), sec.5.1, and the *Model Performance*, sec.5.2.

5.1 Scanning Tunneling Spectroscopy

The STS spectra for the 4ML Bi(110) heights were shown in fig.7 taken from different areas of the sample. It can be easily seen that each 4ML island shown in the inset images 1), 2), 3) and

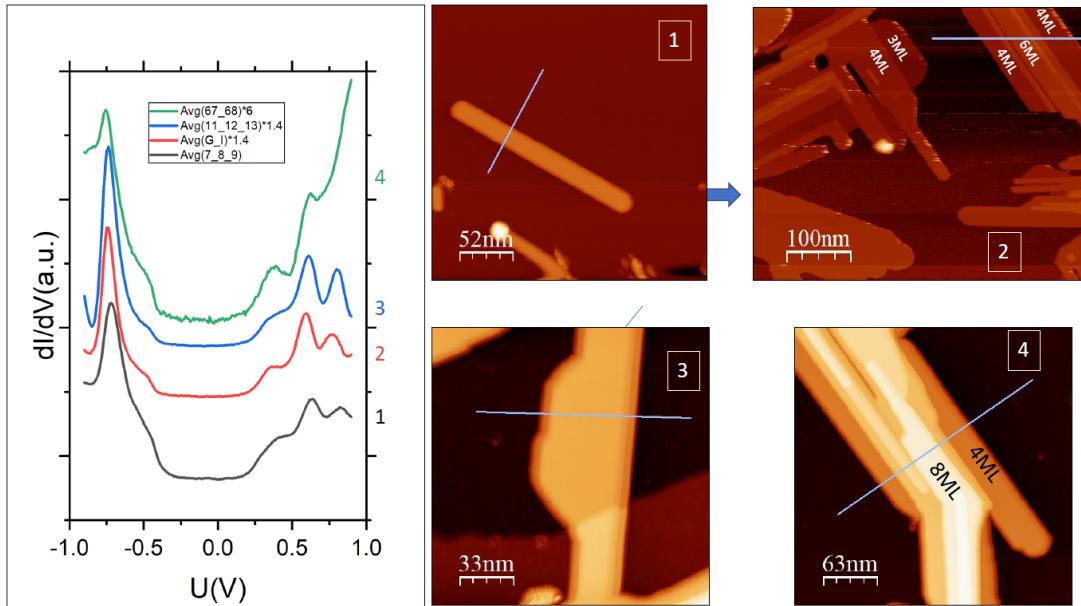


Figure 7: *Proof of concept* that each 4ML island has the same spectra irrespective of any other parameter. Image settings: 100mV, 200pA.

4) in fig.7 has the same differential conductance(dI/dU) spectra. The same behaviour was observed for the island height with different no. of ML, but this independence of the dI/dU spectra w.r.t parameters other than height has only been shown for the 4ML island in fig.7

for the sake of convenience and reproducibility. A further proof of the sole correlation of the

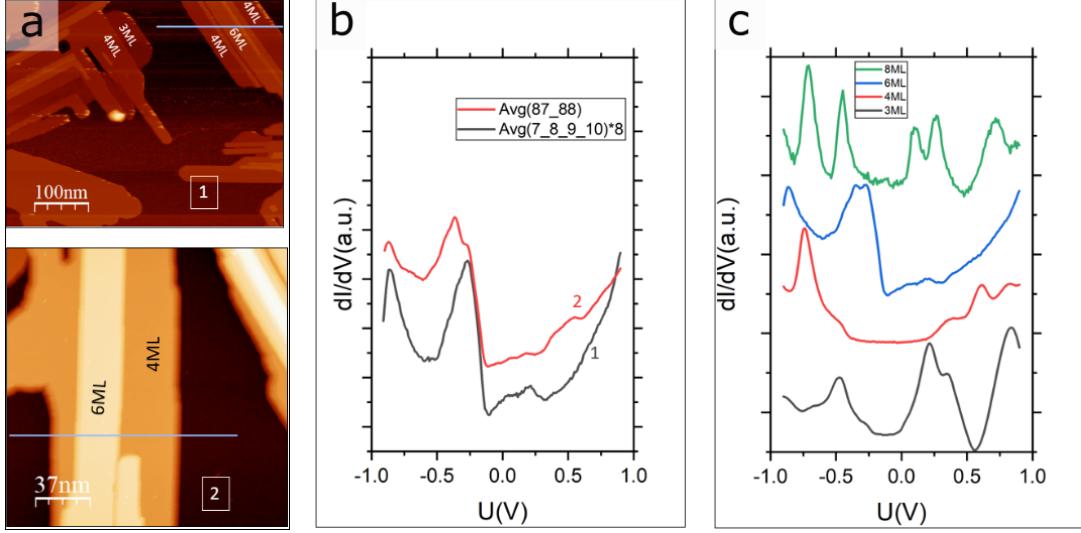


Figure 8: (a) Bi(110) islands growing on different areas of the sample shown in insets 1) and 2). (b) Proof of concept showing reproducibility of 6ML dI/dU spectra taken from insets 1)(black) and 2)(red) of fig.8a. (c) dI/dU spectra of the 3ML, 4ML, 6ML and 8ML islands showing the variance with height. Image settings: 100mV, 200pA.

dI/dU spectra on island height can be seen in fig.8b where the spectra are taken from 6ML islands in different areas of the sample seen in fig.8a. The visible reproducibility of the spectra cannot be understated. It is well understood that the dI/dU spectra scale with the LDOS and hence is a direct evidence of the electronic structure of the Bi(110) under examination[8]. Of particular importance is the spectra shown in fig.8c wherein in the evolution of the LDOS with Bi(110) island height is visible. It can be seen that, other than the 4ML islands all other islands have electronic states crossing the fermi potential located at $U = 0V$. This means that other than the 4ML islands, each of the other island heights have a metallic behaviour and that the thin Bi(110) phases go through a semimetal-semiconductor transition at the 4ML island height. This tunnel-gap/transition seen at the 4ML accounts for the extended thermodynamic stability of the 4ML islands and may well explain the increased probability of sampling 4ML islands, despite the increased deposition flux of Bi as shown in the histograms of fig.5a and fig.5b.

5.2 Model Performance

In order to test the performance of the optimization problem in totality, a test structure called *base structure* was created with known values of $\vec{v} = \{3, 5, 7, 9, 14, 17, 0, 5, 8, 0\}$ as shown in fig.9. Subsequently gaussian noise with $\mu = 0$ and varying σ was added to it and the convergence of the algorithm was tested and the obtained values of the \vec{v} were plotted as frequency distributions for each parameter v_i . The code used to develop the *base structure* is explained in sec.7, listing.2. As can be seen in plots of fig.10a and fig.10b, the σ values for the gaussian noise added to the *base structure* of fig.9 was started at $\sigma = 0.1$ and increased iteratively in steps of $\Delta\sigma = 0.1$ till a value of $\sigma = 3.2$ was reached, as can be seen in fig.10b. The entirety of

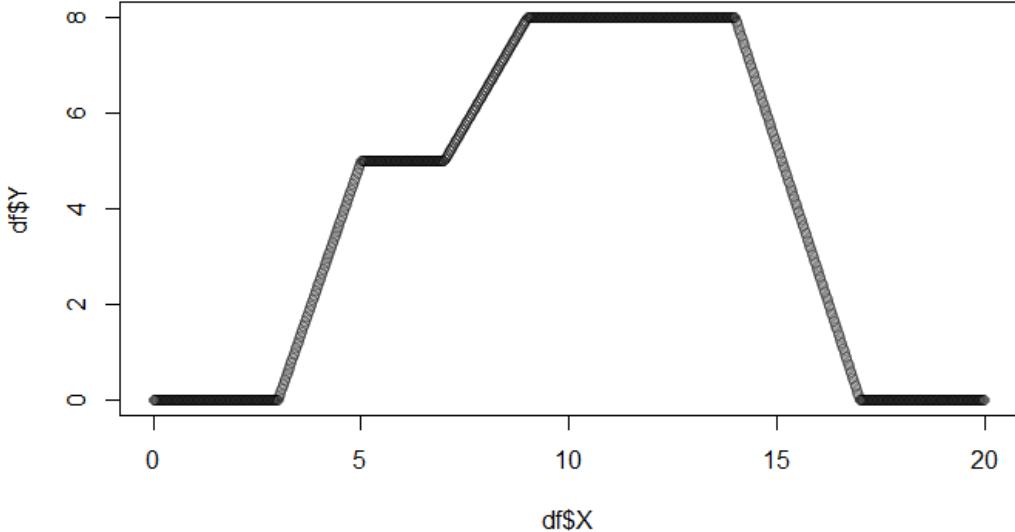


Figure 9: Plot of the *base struct* dataset developed for testing the optimization process. The plots with varying gaussian noise added to this structure can be seen in fig.10.

the 32 plots were divided into two individual plots in (4×4) graphics matrices to allow a better visualization of the effect of increasing σ along with the *approx/guess* structure, shown in red, and the *optimized* structure shown in blue. Ease of visualization of the test was the main reason as to why the entire testing with gaussian noise was subdivided into two subfigures, fig.10a and fig.10b.

It should be noted that for each of the 32 tests shown in fig.10, the starting values of \vec{v}_θ was set at $\vec{v}_\theta = \{2.5, 5, 6.5, 8, 12.5, 16.5, 0, 4, 7, 0\}$. It can be verified from the *approx* structures(shown in red) in fig.10, that the structure is invariant w.r.t σ . Apparent changes might be due to the fact that the y-scaling changes because the values of $\min(\vec{Y})$ and $\max(\vec{Y})$ suffer changes as a result of the randomization of the gaussian noise itself and also the spread of the noise brought about by a change in σ . For example, the y-scale for $\sigma = 3.2$ is contained within the range $\approx (-10, 15)$ (see fig.10b), while that of $\sigma = 0.1$ is contained within the range $\approx (0, 8)$ (see fig.10a). However the *approx* models in red defined by the function $g(x, \vec{v}_\theta)$ of eq.13 are the same since they are only dependent on \vec{v}_θ (see sec.3.)

According to the definitions established in eq.4 through eq.6 in sec.3, with our current vector \vec{v}_θ , $L = |\vec{v}_\theta| = 10$, hence $n = 6$. Thus, the first six elements of the returned vector, \vec{v} , were collected separately as $BP_s(breakpoints)$ and the last four elements separately as $H_s(heights)$ and their distribution of values were plotted. Every plotted distribution, hence adhered to a data-table of 32 entries for each of the σ values shown in fig.10. These plotted distributions can be seen in fig.11a for the *breakpoints* and fig.11b for the *heights*.

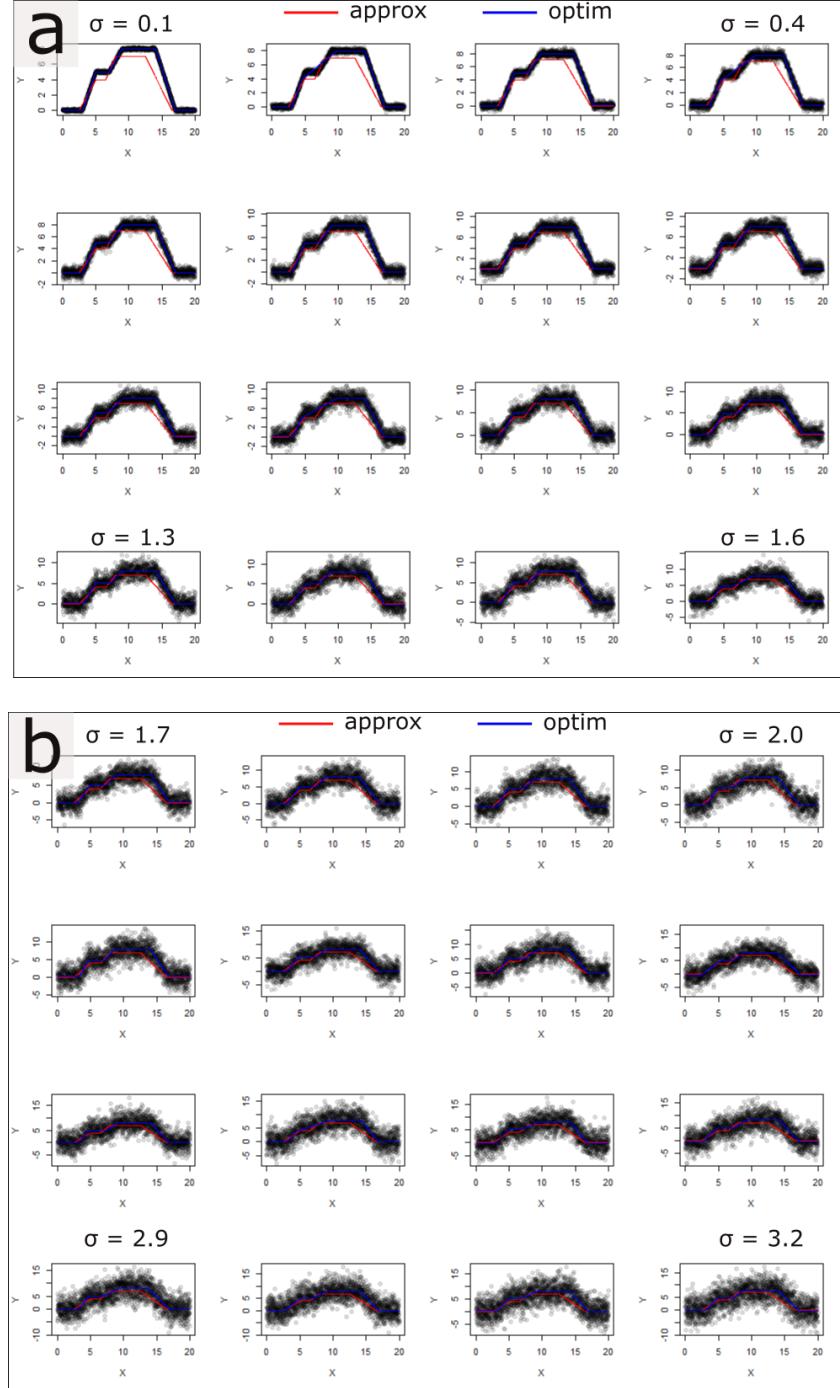


Figure 10: (a) Adding gaussian noise to the test structure starting with $\mu = 0$, $\sigma = 0.1$ and increasing σ from left to right in steps of $\Delta\sigma = 0.1$. The last(bottom right) structure has gaussian noise with $\mu = 0$, $\sigma = 1.6$. (b) Adding gaussian noise to the test structure starting with $\mu = 0$, $\sigma = 1.7$ and increasing σ from left to right in steps of $\Delta\sigma = 0.1$. The last(bottom right) structure has gaussian noise with $\mu = 0$, $\sigma = 3.2$.

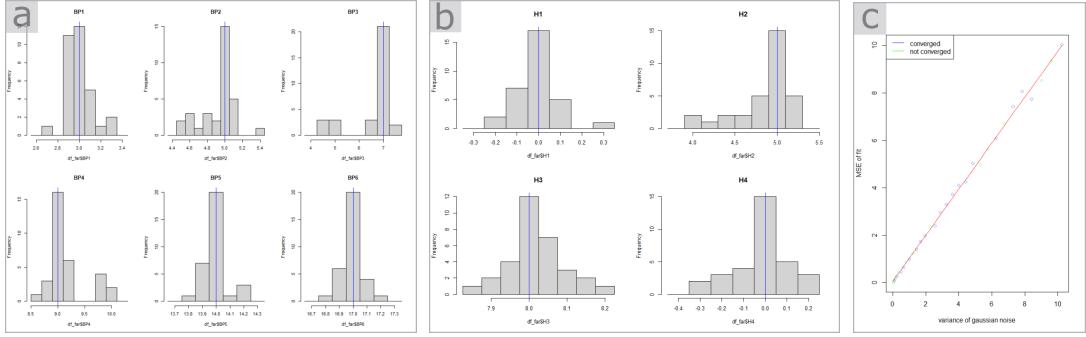


Figure 11: (a) Distribution of BP_j , for $j = [1, 6] \cap \mathbb{Z}$ extracted for values of σ in fig.10. (b) Distribution of H_i , for $i = [1, 4] \cap \mathbb{Z}$ extracted for values of σ in fig.10. (c) Mean squared error(MSE) of *optim* fits of fig.10 plotted against the variance(σ^2) of the gaussian noise added to the *base structure* shown in fig.9.

The vertical blue lines through each histogram, in fig.11, are the values of that particular parameter, BP_j or H_i , used to define the *base structure* of fig.9. It can be seen that, as a result of the optimization, the distributions of the parameter values \vec{v} , returned as a result of the minimization of the loss function $\mathcal{L}(\vec{v}, \vec{v}_\theta, \vec{X}, \vec{Y})^2$ are *qualitatively* distributed around their values defined in the *base structure* of fig.9.

To be able to quantify the model performance for each of the features, table.1 was constructed with every element of the returned vector \vec{v} as shown. The *actual value* refers to the value of the element used to construct the *base structure* of fig.9. The following columns of *mean* and *standard deviation* are derived from the distributions shown in fig.11. The last column of *relative error* was created from the difference of values between the *actual value* and the *mean*.

v_i	feature	actual value	mean	standard deviation	relative error
v_1	BP_1	3	2.99	0.12	0.33%
v_2	BP_2	5	4.92	0.20	1.60%
v_3	BP_3	7	6.57	0.89	6.14%
v_4	BP_4	9	9.14	0.37	1.56%
v_5	BP_5	14	13.99	0.09	0.07%
v_6	BP_6	17	17.00	0.08	0.00%
v_7	H_1	0	-0.02	0.10	NA
v_8	H_2	5	4.84	0.34	3.20%
v_9	H_3	8	8.02	0.07	0.25%
v_{10}	H_4	0	-0.02	0.13	NA

Table 1: Performance table created from values of the distributions obtained in fig.11.

It can be directly seen that the value of the *relative error* for feature BP_3 is higher than the rest and this behaviour could be attributed to the high noise introduced in the vicinity of

²The loss function was defined as $\mathcal{L}(\vec{v}, \vec{X}, \vec{Y})$ in sec.3, but in reality since the final results of the optimization technique are dependent on the initial values of the parameters from where the optimization starts, hence \vec{v}_θ was added into the argument list of the loss function.

BP_3 as a result of increasing σ of the gaussian noise. This phenomena is also visible in the behaviour of the *optim* fits of fig.10. However, it is also interesting to note that in fig.11c, the MSE of the *optim* fits follow a straight line w.r.t the σ^2 of the added gaussian noise, the exact equation for the fit being $MSE = 0.97\sigma^2 + 0.05$. This unity gain linearity of the fit is obvious w.r.t σ^2 , since in this case everything that does not fit to the *optim* structure can be described by the artificially introduced noise whose variance is σ^2 and the MSE is essentially a measure of the unexplained variance in the regression model. However, it is interesting to note that the fit does not deviate from its linear and unity gain feature for higher σ till the limit of $\sigma = 3.2$, hence proving that till this limit the symmetric noise still allows the optimization to relax to a said minima and the model does not collapse.

Another benchmark set for the model till the σ limit of 3.2 can be seen in fig.11c as datapoints differently coloured in green and blue respectively. The blue datapoints are the points for which the model converged, while for the green datapoints the model failed to find convergence. A benchmark called *Convergence accuracy* was empirically created and the value was measured at 62.5%. Hence, around 63% of all fits with the said noise variations along with the said starting \vec{v}_θ converged.

6 Conclusion

In conclusion, statistical mapping of surface structures of the Bi(110)/EG system revealed a preference for the 4ML structure of Bi(110) islands as seen in fig.5a and fig.5b. This preference for the 4ML structure has been shown to come about with a semimetal-semiconductor transition which opens a bandgap and stabilizes the structure at least from an electronic perspective as seen in fig.8c. It has been seen directly that the existence of the squarish Bi(110) lattice itself rejects the geometry of the substrate having any relatively profound influence on the formation of these structures. Whether the gap-opening seen on the 4ML Bi(110) is the result of a correlation effect can be debated and of-course cannot be directly interpreted from the dataset. However, arguments rejecting the influence of substrate geometry extend into the realms of low interaction with the substrate and it has been shown in this work that although the Bi/EG system prefers even no. of MLs as its self assembly structure, the Bi/HOPG system prefers odd no. of MLs as their statistical favourites (fig.6c). Hence, the arguments evolving from "no role of geometry" to "weak interaction of substrate" seems to be a viciously simple one with little hindsight. Of course the real physics behind the estimation of substrate interface effects is an extremely complex one and one that relies mostly on creative theoretical inputs because of the lack of experimental knobs that can be turned to create and design intelligent experiments to explore these subtle effects. Another such avenue which is heavily theory reliant is the existence of even ML structures on the 4ML itself. It can be seen in fig.5b and fig.5c that the growth of Bi(110) structures on the 4ML base extends by an even no. of MLs. The BP-like phase aruments take precedence in this regard [1, 2], but as usual, the existence of these BP-like phases on both HOPG and EG(see fig.6c) call into question again the underlayer-overlayer argumentative methodology. If the HOPG base substrate has a 3ML base growth structure, while the EG has a 4ML base growth structure, why wouldn't there be different interface effects for the next BP-like 2ML phase, and if so, why don't we have a BP-like 2ML phase directly on graphene but only on top of previously grown Bi islands. By all accounts of linear reasoning interactions of Bi with Bi need to be higher than that of Bi with graphene, and hence the BP-like phase should get transformed either on the 3ML Bi/HOPG structure or the 4ML Bi/EG structure. At this point it should

suffice to say that the progress of science is spearheaded only by the painstaking erradication of previously accepted hypothesis, built on seemingly ignorant premise. In this respect, the limits of experimental work should be respected and better algorithms and chronological methods are needed to further increase the precision of the experimental conclusion to ensure the rhythmic dissolution of previously coveted, yet erroneous notions of reality.

7 Appendix

7.1 The Optimization Problem

This section is dedicated to explaining the code which was used to formaulate and execute the optimization problem in sec.3. The entire enactment of the code shown in listing.1 is available as a package on github in the repository *StatsChitran* under the author name *Chitran1987*. The direct link to the repository is presented here.

```

1 chng_pnt_algo<-function(v,dat, namX='X', namY='Y', col_pt=rgb(0,0,0,0.125), col
2   _app='red', col_op='blue', gr_app=T, gr_op=T, gr_leg=T){
3   names(dat)<-c('V1', 'V2')
4   L<-length(v)
5   Bp<-v[seq(1,(2*(L-1)/3))]#####number of breakpoints
6   H<-v[seq(((2*L+1)/3),L)] #####number of plateau heights
7
8   #######################################################
9   ##########creating the breakpoint height dataframe#####
10  Ht<-vector(mode = 'numeric', length=length(Bp))
11  ##########creating the Htvalues for the breakpoints#####
12  k<-2
13  for (i in 1:length(Bp)) {
14    if(i==1){
15      Ht[i]<-H[i]
16    }
17    else if(i==length(Bp)){
18      Ht[i]<-H[length(H)]
19    }
20    else{
21      if(i%%2==0){
22        Ht[i] <- H[k]
23        k <- k+1
24      }
25      else{
26        Ht[i]<-Ht[i-1]
27      }
28    }
29  ##########
30  ##########creating the dataframe#####
31  Bp_H_df<-data.frame(matrix(c(Bp,Ht), byrow = F, ncol = 2))
32  names(Bp_H_df)<-c('V1','V2')
33  ##########
34  ##########
35  ##########
36  ##########
37  ##########
38  #####
39  ##########
40  ##########create a function for accepting the profile that you want to create
41  prof_struct<-function(v,dat){
42    tgt<-dat
43    L<-length(v)

```

```

44     Bp<-v[seq(1,(2*(L-1)/3))]#####number of breakpoints
45     H<-v[seq(((2*L+1)/3),L)] #####number of plateau heights
46     for (i in 1:length(Bp)) {
47       if(i==1){
48         tgt$V2[tgt$V1<=Bp[i]]<-H[i]
49       }#if close
50       else if(i==length(Bp)){
51         tgt$V2[tgt$V1>=Bp[i]]<-H[length(H)]
52         tgt$V2[tgt$V1>=Bp[i-1] & tgt$V1<=Bp[i]]<-((H[((i/2)+1]-H[i/2])/(Bp[i]-Bp
53 [i-1]))*(tgt$V1[tgt$V1>=Bp[i-1] & tgt$V1<=Bp[i]]-Bp[i-1])+H[i/2]
54       }#else if close
55       else{
56         if(i%%2==0 ){
57           tgt$V2[tgt$V1>=Bp[i-1] & tgt$V1<=Bp[i]]<-((H[((i/2)+1]-H[i/2])/(Bp[i]-
58 Bp[i-1]))*(tgt$V1[tgt$V1>=Bp[i-1] & tgt$V1<=Bp[i]]-Bp[i-1])+H[i/2]
59         }#if close
60         else{
61           tgt$V2[tgt$V1>=Bp[i-1] & tgt$V1<=Bp[i]]<-H[((i+1)/2)
62         }#small else close
63       }#bigelse close
64     }#forloopclose
65     return(tgt)
66   } #function close
67 ##########
68 #####define the plot limits#####
69 xmin <- min(dat$V1)-(1/4)*(max(dat$V1)-min(dat$V1))
70 xmax <- max(dat$V1)+(1/4)*(max(dat$V1)-min(dat$V1))
71 xl <- c(xmin,xmax)
72 ymin <- min(dat$V2)
73 ymax <- max(dat$V2)+(1/4)*(max(dat$V2)-min(dat$V2))
74 yl <- c(ymin, ymax)
75 #####
76 #####first plot#####
77 plot(dat$V1, dat$V2, xlab = namX, ylab = namY, col=col_pt, pch=19)
78 dat1<-prof_struct(v,dat)
79 dat1<-rbind(dat1, Bp_H_df)
80 dat1<-dat1[order(dat1$V1),]
81 if(gr_app==T){
82   lines(dat1$V1,dat1$V2, col=col_app)
83 }
84
85
86 #####setup the constraint matrix and the vector
87 #####number of points#####number of constraints=n-1(for breakpoints)+2*((length
88 (v)+2)/3)(for the plateau heights)
89 n<-2*(length(v)-1)/3 #####number of breakpoints
90 #####constraint matrix
91 m<-matrix(0,nrow = 2*n+1, ncol = length(v))
92 #####loop for the n-1 breakpoint constraints
93 for (i in 1:n-1) {
94   m[i,i]<-1
95   m[i,i+1]<-1
96 }
97 #####loop for the 2*((length(v)+2)/3) plateau height constraints
98 for(i in n:(n+((n/2)+1)-1)){
99   m[i,i+1]<-1
100 }
101 for(i in ((n+((n/2)+1)-1)+1):(2*n+1) ){
102   m[i,i-((n/2)+1)]<-1

```

```

103 }
104 #####constraint vector
105 #####length of constraint vector is equal to no. of constraints
106 b<-rep(0L, 2*n+1)
107 #####use zeros for the breakpoint constraints
108 for(i in 1:n-1){
109   b[i]<-0
110 }
111 for (i in n:(n+((n/2)+1)-1)) {
112   b[i]<-min(dat$V2)
113 }
114 for (i in ((n+((n/2)+1)-1)+1):(2*n+1)) {
115   b[i]<--max(dat$V2)
116 }
#####write down the objective function#####
117 f<-function(v){
118   dat2<-dat
119   tgt2<-prof_struct(v,dat2)
120   e<-sum((tgt2$V2-dat$V2)^2)
121   return(e)
122 }
123 r<-constrOptim(v,f,NULL,m,b)
124
#####calculating the return values#####
125 v<-r$par
126 L<-length(v)
127 Bp<-v[seq(1,2*(L-1)/3)]
128 H<-v[seq((2*L+1)/3, L)]
#####
129 #####creating the breakpoint height dataframe#####
130 Ht<-vector(mode = 'numeric', length=length(Bp))
131 #####creating the Htvalues for the breakpoints#####
132 k<-2
133 for (i in 1:length(Bp)) {
134   if(i==1){
135     Ht[i]<-H[i]
136   }
137   else if(i==length(Bp)){
138     Ht[i]<-H[length(H)]
139   }
140   else{
141     if(i%%2==0){
142       Ht[i] <- H[k]
143       k <- k+1
144     }
145     else{
146       Ht[i]<-Ht[i-1]
147     }
148   }
149 }
150 }
151 }
152 }
153 #####
154 #####creating the dataframe#####
155 Bp_H_df<-data.frame(matrix(c(Bp,Ht), byrow = F, ncol = 2))
156 names(Bp_H_df)<-c('V1','V2')
157 #####
158 #####plotting the optimized function#####
159 tgt<-prof_struct(v,dat)
160 tgt<-rbind(tgt, Bp_H_df)
161 tgt<-tgt[order(tgt$V1),]
162 if(gr_op==T){
163

```

```

165   lines(tgt$V1,tgt$V2, col=col_op)
166 }
167 if(gr_app==T & gr_op==T & gr_leg==T){
168   legend('topleft', legend = c('approx', 'optim'), col = c(col_app, col_op),
169   lty=1:1)
170 }
171 #####convergence factor#####
172 con<-vector(mode = 'character', length = 1)
173 if(r$convergence==0){
174   con<-'solution has converged'
175 }
176 else{
177   con<-'solution has not converged'
178 }
179 #####
180 #####result of final convergence#####
181 final<-list(return=r,Optimized_Vector=v, BreakPoints=Bp, Heights=H,
182   Convergence=con)
183 return(final)
184 }
```

Listing 1: code for the optimization problem described in sec.3.

Lines 1 through 5 split up the input vector \vec{v}_θ into $\vec{B}\vec{P}$ and \vec{H} vectors, see eq.10. Lines 7 through 30 map each element in $\vec{B}\vec{P}$ to an element in \vec{H} . At the end of line 30, a vector $\vec{H}t$ is created with the same cardinality as $\vec{B}\vec{P}$. Since in the lines mentioned above, $f : \vec{B}\vec{P} \rightarrow \vec{H}t$, is defined as a bijection. In lines 31 through 34, a two-column dataframe is set up named, Bp_H_df . This is possible because of the bijection defined above. Lines 41 through 64 are used to define the $g(x, \vec{v}_\theta)$ function in eq.13 using the *prof_struct* function. Lines 67 through 84 are used to overlay the *approx* dataset $g(x, \vec{v}_\theta)$ over the dataset in interest. The *approx* dataset is always shown in red³, see fig.1b, fig.10a, fig.10b. Lines 87 through 116 have been used to setup the constraint matrix and vectors. Here $[m]$ is the constraint matrix initiated in line 91, whereas $[b]$ is the constraint vector initiated in line 106. It can be seen that $[m]$ has dimensions $(2n + 1 \times L)$, whereas vector $[b]$ has dimensionality $(2n + 1 \times 1)$. The constraint matrix-vector pedagogy for the R package *constrOptim* has been shown in eq.19.

$$[m] \cdot [v] \geq [b] \quad (19)$$

Here, $[v]$ is the matrix representation of \vec{v} , hence a $(L \times 1)$ matrix. These $2n + 1$ constraints were created as follows. First $n - 1$ constraints used to satisfy eq.16, see lines 91 through 96. Next, for eq.17, we need $2(L - n)$ ⁴ constraints. Since $2(L - n) = n + 2$ in accordance to eq.6, hence the next constraints were constructed in lines 97 through 103 for the *matrix* $[m]$ side and lines 111 through 116 for $[b]$.

Lines 117 through 123 were used to construct the objective function $\mathcal{L}(\vec{v}, \vec{X}, \vec{Y})$, while line 124 is where the real optimization takes place using the *constrOptim* call. Lines 127 through 130 segregates the returned vector \vec{v} into $\vec{B}\vec{P}$ and \vec{H} as before and then 134 through lines 154, the optimized bijection $\vec{H}t$ is created. The optimized dataframe *Bp_H_df* was created in lines 156 through 159, followed by creating the final optimized dataframe to be plotted in

³The red colour is the default value of the *col_app* argument for the *chng_pnt_algo* function. It can be changed when calling the function and redefined by the user. However, for the purposes of this work, *col_app* has been left at its default setting.

⁴The factor 2 is multiplied to take care of both the min and the max conditions

lines 161 through 163. Lines 164 through 170 plots these optimized data frame provided the conditions gr_op and gr_app are set to *TRUE*. The *optim* dataframe is plotted in blue for the purposes of the current work, see fig.10a, fig.10b, fig.1b. Lines 171 through 179 work to return the convergence criterion of the optimization process wheres in lines 182 and 183 the return datatype is defined. It is a list comprising of the return from the *ConstrOptim* optimization in line 124, the vector \vec{v} , the Breakpoint vector \vec{BP} , the heights vector \vec{H} and the convergence criterion.

7.2 The Base Structure

```

1 rm(list=ls())
2
3 ##create the test dataset
4 X <- seq(0, 20, by=0.01)
5 Y <- rep(NA, times = length(X))
6 df <- data.frame(X,Y)
7 #plot(df$X, df$Y)
8 ClearPlot()
9 df$Y[df$X <= 3] <- 0
10 df$Y[df$X >= 3 & df$X <=5] <- lineval(c(3,0), c(5,5), X = df$X[df$X>=3 & df$X
11 <=5])
12 df$Y[df$X >= 5 & df$X <=7] <- 5
13 df$Y[df$X >= 7 & df$X <=9] <- lineval(c(7,5), c(9,8), X = df$X[df$X>=7 & df$X
14 <=9])
15 df$Y[df$X>=9 & df$X<=14] <- 8
16 df$Y[df$X>=14 & df$X<=17] <- lineval(c(14,8), c(17,0), X = df$X[df$X>=14 & df$X
17 <=17])
18 df$Y[df$X>=17] <- 0
19 plot(df$X, df$Y, type = 'l')
20 write.table(df, 'D:/SynologyDrive/Didos_self/Career/Future/Math_Stats/ML_
lectures/Term_Paper/model_performance/base_structure/base_struct.txt', sep
= '\t', row.names = F)

```

Listing 2: code for the base structure in fig.9.

In this section the code used in listing.2 is explained. This was used to develop the structures to which noise was added and studies were conducted in sec.5. As can be seen, in lines 1 through 6, a dataframe was created which reamained empty w.r.t the independent varaiable. In lines 11 through 15, the independent variable, dfY$ was filled up with values adhering to $\vec{v} = \{3, 5, 9, 14, 17, 0, 5, 8, 0\}$. The last *breakpoint* from \vec{v} being 17, while the last *Height* being 0, hence the attribution in line 15. Lines 16 and 17 are just used to plot the data and write the data to a certain location.⁵

References

- [1] KOWALCZYK, P. J.; MAHAPATRA, O; McCARTHY, D.; KOZLOWSKI, W; KLUSEK, Z, and BROWN, S.: "STM and XPS investigations of bismuth islands on HOPG", Surface science, **605.7-8**, 659–667, (2011).
- [2] NAGAO, T.; SADOWSKI, J.; SAITO, M.; YAGINUMA, S.; FUJIKAWA, Y.; KOGURE, T.; OHNO, T.; HASEGAWA, Y.; HASEGAWA, S., and SAKURAI, T.: "Nanofilm Allotrope and Phase Transformation of Ultrathin Bi Film on Si(111)-7×7", **93**.
- [3] AYUELA, A.; OGANDO, E., and ZABALA, N.: "Quantum size effects of Pb overlayers at high coverages", Applied Surface Science, **254.1**, 29–31, (2007).

⁵The *ClearPlot()* function used in line 8, listing2 can be found in the *StatsChitran* package in github under the author name *Chitran1987*.

- [4] AYUELA, A.; OGANDO, E., and ZABALA, N.: "Restored quantum size effects of Pb overlayers at high coverages", **75**.**15**, 153403, 153403, (2007).
- [5] PAN, S.; LIU, Q.; MING, F.; WANG, K., and XIAO, X.: "Interface effects on the quantum well states of Pb thin films", Journal of Physics Condensed Matter, **23**.**48**, 485001, 485001, (2011).
- [6] SLOMSKI, B.; MEIER, F.; OSTERWALDER, J., and DIL, J. H.: "Controlling the effective mass of quantum well states in Pb/Si(111) by interface engineering", Physical Review B, **83**.**3**, 035409, (2010). arXiv: 1010.5371 [cond-mat.mtrl-sci].
- [7] KUNDU, R.; MISHRA, P.; SEKHAR, B. R.; MANIRAJ, M., and BARMAN, S. R.: "Electronic structure of single crystal and highly oriented pyrolytic graphite from ARPES and KRIPES", Physica B Condensed Matter, **407**.**5**, 827–832, (2012).
- [8] TERSOFF, J. and HAMANN, D. R.: "Theory and Application for the Scanning Tunneling Microscope", **50**, 1998–2001,
- [9] KRUSKOPF, M. et al.: "Comeback of epitaxial graphene for electronics: large-area growth of bilayer-free graphene on SiC", 2D Materials, **3**.**4**, 041002, 041002, (2016). arXiv: 1606.01709 [cond-mat.mtrl-sci].
- [10] KRÖGER, P.; ABDELBAREY, D.; SIEMENS, M.; LÜKERMANN, D.; SOLOGUB, S.; PFNÜR, H., and TEGENKAMP, C.: "Controlling conductivity by quantum well states in ultrathin Bi(111) films", **97**.**4**, 045403, 045403, (2018).