Machine Learning Model Knowledge based forGNR Interconnects

By

Abhay Chaurasiya[2018IMT-005]

Under the supervision of:

Dr. Somesh Kumar, Ph.D.
Assistant Professor,
ABV-Indian Institute of Information Technology and Management Gwalior(IIITM),
Gwalior, Madhya Pradesh-474015

Outline

- In this Project we made various Machine Learning Models for predicting MFP due to scattering by edge roughness at given wavelength , predicting MFP due to scattering by edge roughness at given fermi energy ,predicting the effective MFP of electrons in GNR at given wavelength ,predicting the effective MFP of electrons in GNR at given fermi energy , predicting the number of layers in MLGNR at given thickness, predicting scattering resistance , lumped resistance and quantum resistance resulting in the prediction of the total resistance.

Introduction

- With the scaling down of integrated circuits (ICs) into nanometer sizes, new challenges have emerged for conventional Cu interconnects.
- These Problems are due to surface scattering, heat dissipation, electromigration, grain boundaries, and current capacity issues .
- Carbon nanomaterials like graphene nanoribbons (GNRs) and carbon nanotubes (CNTs) have attracted a better research interest as candidates for replacing Cu nanowire interconnects in advanced ICs.
- Multilayer graphene nanoribbon MLGNR interconnects performance are better than Cu counterparts with the same dimensions both in terms of bandwidth and time delay.

- Due to its high exceptional strength, low coefficient of thermal expansion and ultra-high thermal conductivity(3000-6000W/m.K) Carbon Nanotube(CNT) is one of the most promising materials in the field of advanced materials.

- On the basis of edge geometry there are three types of GNRs available . They are:
  - Armchair (AGNR)
  - Zigzag (ZGNR)
  - Chiral (CGNR)
- 33% of AGNRs are semiconducting with respect to the number of carbon atoms present in AGNR width and also entire ZGNRs are metallic.

Literature Review

- The on chip GNR interconnects have been examined as one of the most stimulating areas in wide-ranging scale integration for the last several years.
- With the help of Properties of the interconnects we will best describe the reliability and the on chip performance instead of the transistors.
- Due to the extraordinary electrical and their other properties, graphene becomes a dependable contender for next generation interconnects.
- Graphene has the high current density with lowest resistivity ,high electron mobility and large mean free path..

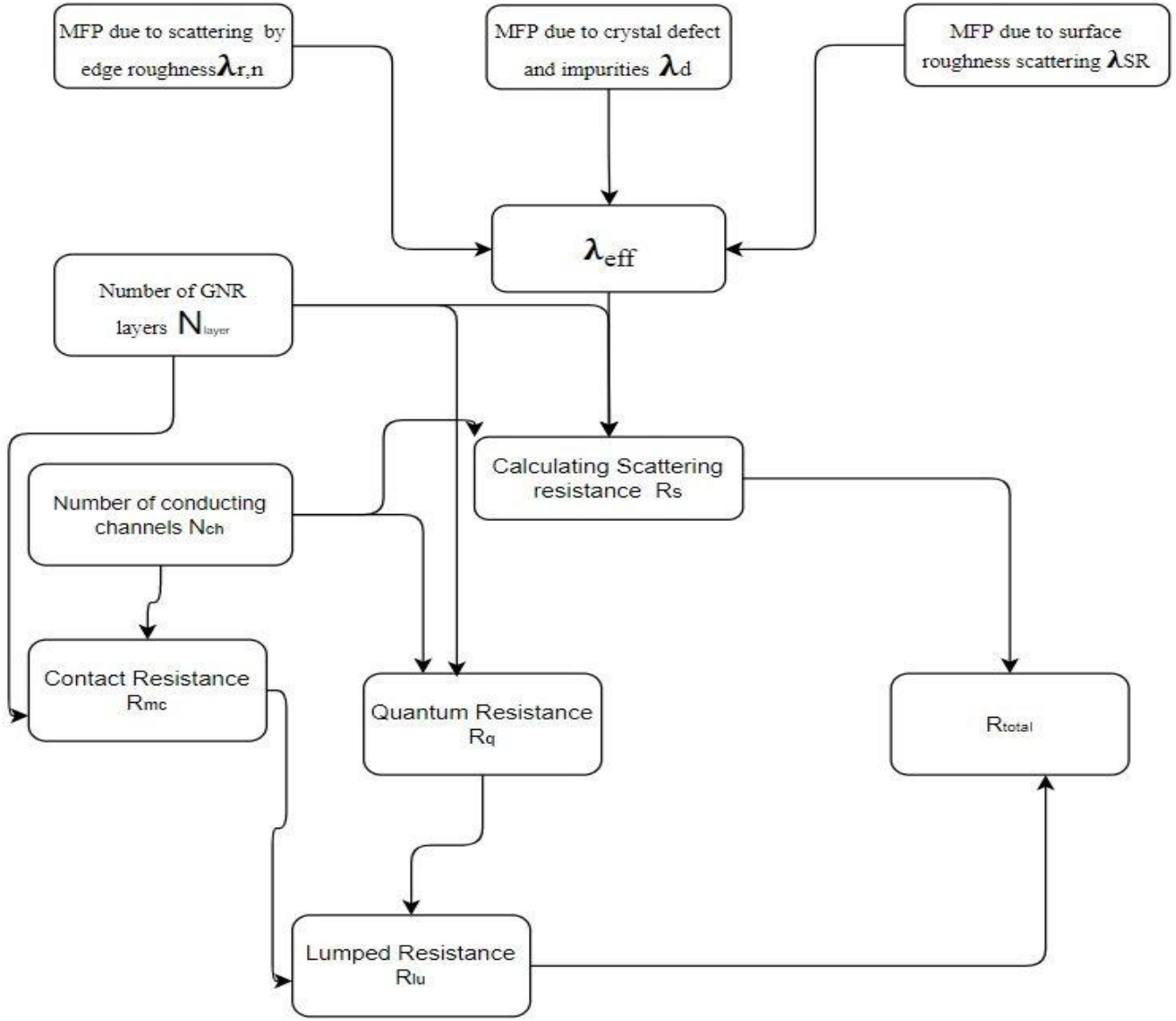
- From the research it has been concluded that narrow width graphene nanoribbon (GNR) or graphene sheet is the most acceptable interconnect material.
- the electrical property of GNR can be changed by changing geometric configuration to a minor range to differentiate the modesty of graphene film.
- The electrical resistance of SLGNR is relatively high as compared to Cu.

Objectives

- Creating a machine learning model to predict the number of layers in GNR at given thickness.
- Developing a machine learning model to predict MFP due to scattering by edge roughness at given wavelength
- Developing a machine learning model to predict MFP due to scattering by edge roughness at given fermi energy
- Developing a machine learning model to predict the effective MFP of electrons in GNR at given wavelength
- Developing a machine learning model to predict the effective MFP of electrons in GNR at given fermi energy
- Developing a machine learning model to predict scattering resistance,lumped resistance and quantum resistance resulting in the prediction of the total resistance.

Methodology

Flow Chart of the work.



Methodology

System Architecture

The total number of GNR layers depends on the MLGNR thickness (T) and interlayer spacing (δ) between adjacent MLGNRs as follows:

$$N_{layer} = 1 + Integer\left(\frac{t}{\delta}\right)$$

The lumped resistance is assumed to be equally divided between the two contacts as follows:

$$R_{lu} = (R_{mc} + R_q)/2$$

The scattering resistance is given as :

$$R_s = 12.9/(N_{ch} * N_{layer} * \lambda_{eff})$$

Methodology

System Architecture

The effective MFP for the nth sub-band $\lambda_{eff,n}$ can be calculated as: $$\frac{1}{\lambda_{eff,n}} = \frac{1}{\lambda_d} + \frac{1}{\lambda_{pn}} + \frac{1}{\lambda_{er,n}} + \frac{1}{\lambda_{sr}}$$

$\lambda_{er,n}$ is given by : $$\lambda_{er,n} = \frac{W}{p}\sqrt{\left(\frac{2WE_f}{nhv_f}\right)^2 - 1}$$

λ_d is the MFP due to crystal defect and impurities which is taken as 1um.

$\lambda_{p,n}$ is the MFP due to electron phonon scattering which is 70um.

λ_{sr} is the MFP due to surface roughness scattering which is given as $\lambda_{sr}=c/y^4_{sr}$ where $c=2.07*10^9$ and y_{sr}=surface roughness which is taken as 20pm.

Implementations

Tools used:

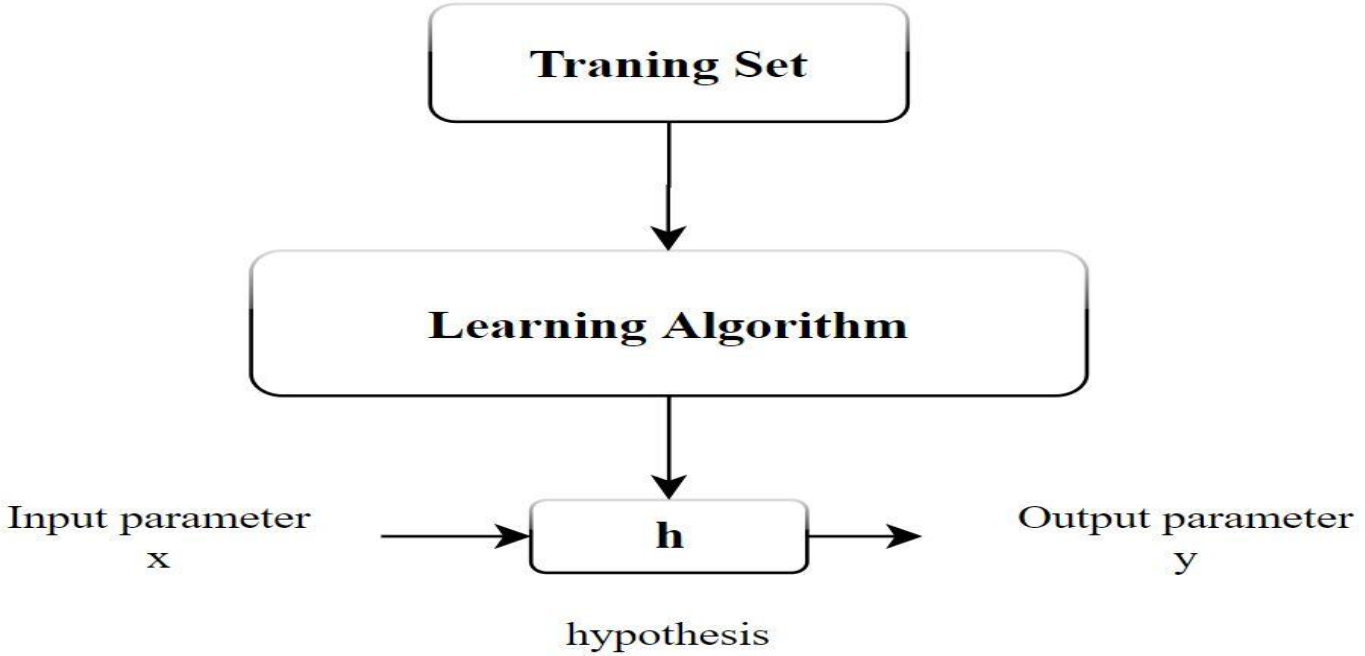
We have used a number of programming and machine learning tools for our project. Some of them are:

- Programming Language - Python
- Libraries used:
  - numpy for creating arrays and working in them.
  - pandas for reading and creating csv files and working on dataframes.
  - sklearn for making use of some predefined models like linear regression and polynomial regression.
  - matplotlib for plotting various graphs and contours.
  - SciPy for extracting the values of several scientific constants used in our project.
- Software for plotting graphs- Origin.
- Platform used for creating models-Jupyter Notebook and Google Collab Notebook.

Implementations

Setup:

- We have used the following setup while creating our each model for each equation:



Implementations

Setup:

- For equation

$$N_{layer} = 1 + Integer\left(\frac{t}{\delta}\right)$$

We have used linear regression model and hypothesis as **h= $\boldsymbol{\theta}_0$ + $\boldsymbol{\theta}_1$*x** where our model is trained to learn the value of $\boldsymbol{\theta}_0$ and $\boldsymbol{\theta}_1$.

How do we train our model ?

- We provide the values of t as x to our model and N_{layer} as y to our model.
- Model tries to learn the value of $\boldsymbol{\theta}_0$ and $\boldsymbol{\theta}_1$.
- Model learns the value of $\boldsymbol{\theta}_0$ and $\boldsymbol{\theta}_1$ and use it to predict the value of N_{layer}.
- We plot the graph of N_{layer} vs t and then between N_{layer} predicted and t to know the difference between our model and actual values.

Error Function used:

We have used the least square error function for our hypothesis.

$$J=(1/n)\Sigma(y-h(\boldsymbol{\theta}_x)$$

We have used gradient descent algorithm to learn the value of $\boldsymbol{\theta}_0$ and $\boldsymbol{\theta}_1$ for our model.

Implementations

Setup:

- For equation between $\lambda_{er,n}$ and E_f , $\lambda_{er,n} = \frac{W}{p}\sqrt{\left(\frac{2WE_f}{nhv_f}\right)^2 - 1}$

We have assumed that $\lambda_{er,n}$=y and E_f=x and taking other terms as constant and squared the equation on both sides . Through which we got $y^2=b*(a*x^2-1)$. Since it is a polynomial equation, we used a polynomial regression model to learn the value of a and b. The hypothesis we have used is $h=b*(a*x^2-1)$ where our model is trained to learn the value of a and b.

How do we train our model ?

- We provide the values of E_f as x to our model and $\lambda_{er,n}$ as y to our model.
- Model tries to learn the value of a and b.
- Model learns the value of a and b and uses it to predict the value of $\lambda_{er,n}$.
- We plot the graph of $\lambda_{er,n}$ vs E_f and then between $\lambda_{er,n}$ predicted and E_f to know the difference between our model and actual values.

We have used the same error function in this model as well.

Implementations

Setup:

- For equation $\frac{1}{\lambda_{eff,n}} = \frac{1}{\lambda_d} + \frac{1}{\lambda_{pn}} + \frac{1}{\lambda_{er,n}} + \frac{1}{\lambda_{sr}}$

  We have used the same linear regression model as discussed above taking $1/\lambda_{er,n}=x$ and $1/\lambda_{eff,n}=y$ making the above equation as $y=m*x+c$.

- For equation between $\lambda_{er,n}$ and w , $\lambda_{er,n} = \frac{W}{p}\sqrt{\left(\frac{2WE_f}{nhv_f}\right)^2 - 1}$

  We have used the same polynomial regression model as discussed above taking $\lambda_{er,n}=y$ and $w=x$ and squaring the equation making the equation as $y^2=a*x^4+b*x^2$ which is a polynomial equation.

Implementations

Techniques Used:

Supervised Learning

In a *supervised* learning task, the data sample would contain a target attribute {y} *y*, also known as *ground truth*. And the task is to learn a function F, that takes the non-target attributes X and outputs a value that approximates the target attribute, *i.e.* F(X) \approx y $F(X) \approx y$. The data with a target attribute is called "*labeled*" data. Based on the above definition, for the task of predicting the given parameters with the labeled data, it can be told it is a supervised learning task.We have used two special cases of supervised learning :

1. Linear Regression:-Linear regression is a basic and commonly used type of predictive analysis. The overall idea of regression is to examine two things: (1) does a set of predictor variables do a good job in predicting an outcome (dependent) variable? (2) Which variables in particular are significant predictors of the outcome variable, and in what way do they–indicated by the magnitude and sign of the beta estimates–impact the outcome variable? These regression estimates are used to explain the relationship between one dependent variable and one or more independent variables. The simplest form of the regression equation with one dependent and one independent variable is defined by the formula y = c + m*x, where y = estimated dependent variable score, c = constant, m = regression coefficient, and x = score on the independent variable.

Implementations

Techniques Used:

Polynomial Regression:-Polynomial regression is a form of regression analysis in which the relationship between the independent variable x and the dependent variable y is modelled as an nth degree polynomial in x. Polynomial regression fits a nonlinear relationship between the value of x and the corresponding conditional mean of y, denoted E(y |x). Although polynomial regression fits a nonlinear model to the data, as a statistical estimation problem it is linear, in the sense that the regression function E(y | x) is linear in the unknown parameters that are estimated from the data. For this reason, polynomial regression is considered to be a special case of multiple linear regression. It is defined by the formula:

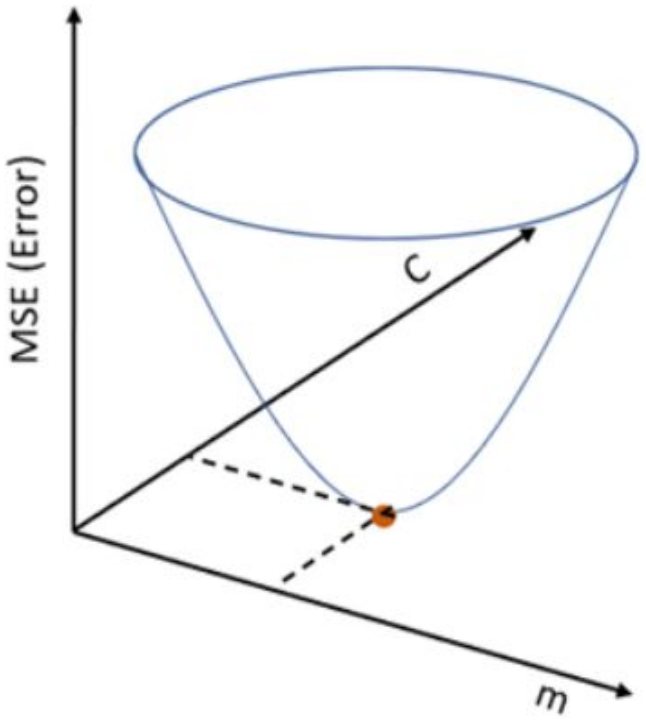
$$y=a*x^{n}+b*x^{n-1}+.....+k*x^{0}$$

which is a polynomial equation of n^{th} degree.

Implementations
Techniques Used:

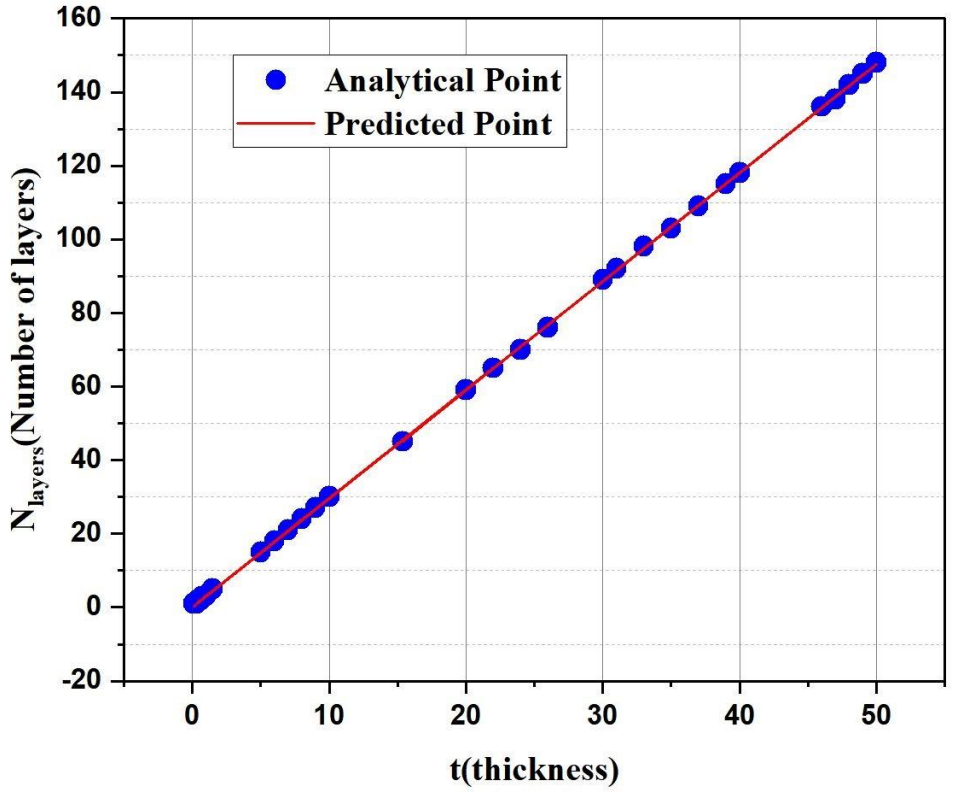
Gradient Descent

We have used the technique of gradient descent for learning the various constants in our models.Gradient descent is a first-order iterative optimization algorithm for finding a local minimum of a differentiable function. The idea is to take repeated steps in the opposite direction of the gradient (or approximate gradient) of the function at the current point, because this is the direction of steepest descent. Conversely, stepping in the direction of the gradient will lead to a local maximum of that function; the procedure is then known as gradient ascent.Itis an algorithm that finds the best-fit line for a given training dataset in a smaller number of iterations.If we plot m and c against MSE, it will acquire a bowl shape as shown :



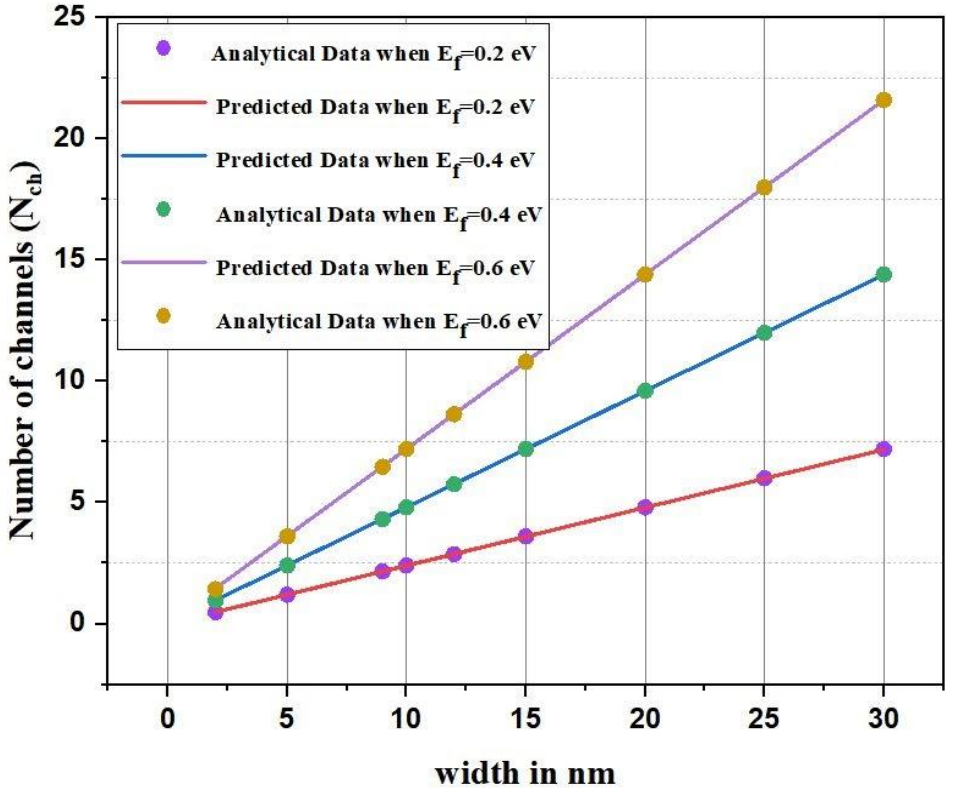
Results and Discussion

We created a linear regression model to predict the number of layers in GNR at a given thickness. In our model , Root mean squared error is 0.18648667982023368 and R score is 0.999933560292709 7which shows the accuracy of our model.The graph shown in Fig 6. Shows us that the number of layers increases as the thickness of GNR is increased.



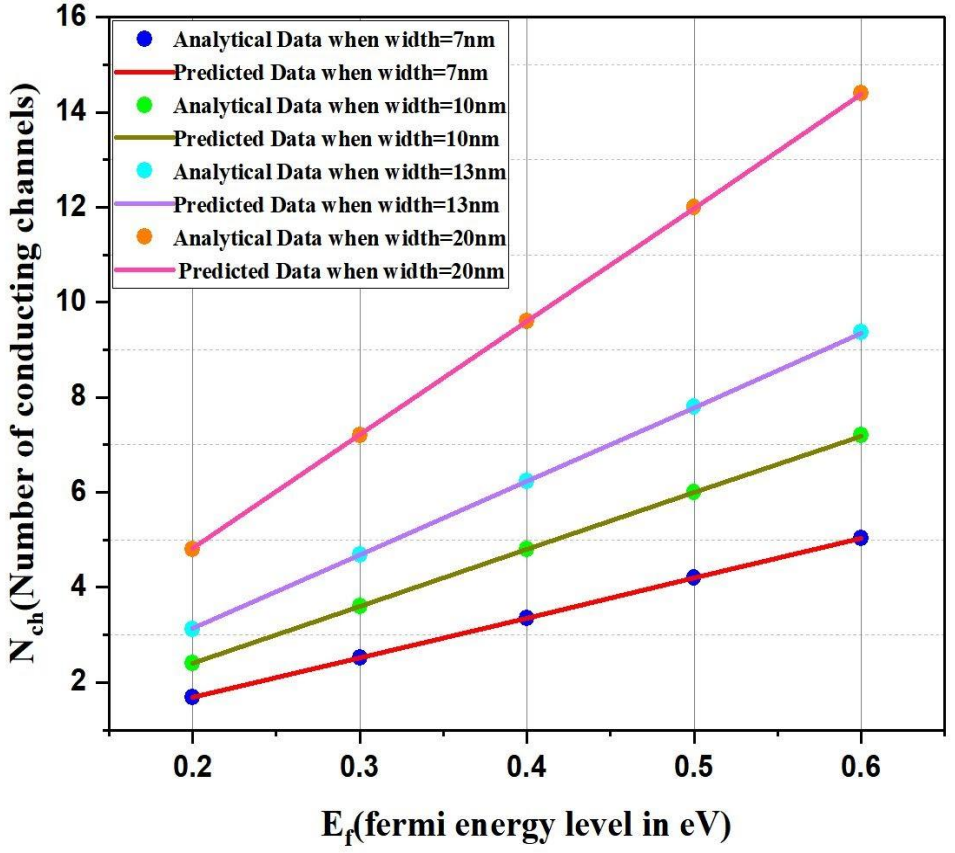
Results and Discussion

We created a linear regression model to predict the number of channels in GNR at a given width and fixed value of fermi energy.In our model, we trained the model for three values of fermi energy which are 0.2eV, 0.4eV and 0.6eV. Root mean squared error is different for different values of fermi energy and R score is 0.999991604629730 4which shows the accuracy of our model.The graph shown in Fig 7. shows us that the number of channels increases as the width of GNR is increased. It also indicates that more is the fermi energy, more will be the value of the number of channels at fixed width.



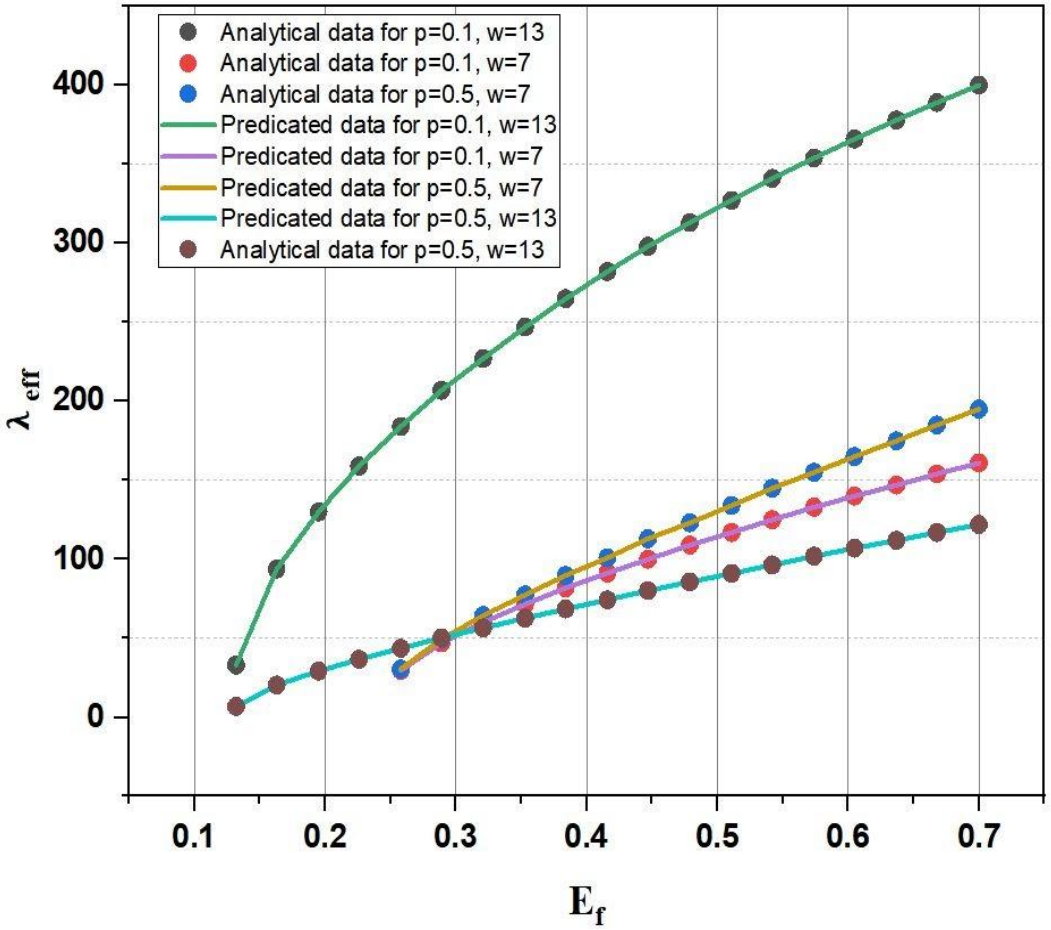
Results and Discussion

We created a linear regression model to predict the number of channels in GNR at a given fermi energy and fixed value of width.In our model, we trained the model for four values of width which are 7nm, 10nm, 13nm and 20nm. Root mean squared error is different for different values of width and R score is 0.9999763167516774which shows the accuracy of our model.The graph shown in Fig 8. shows us that the number of channels increases as the fermi energy of GNR is increased. It also indicates that more is the width, more will be the value of the number of channels at fixed fermi energy.



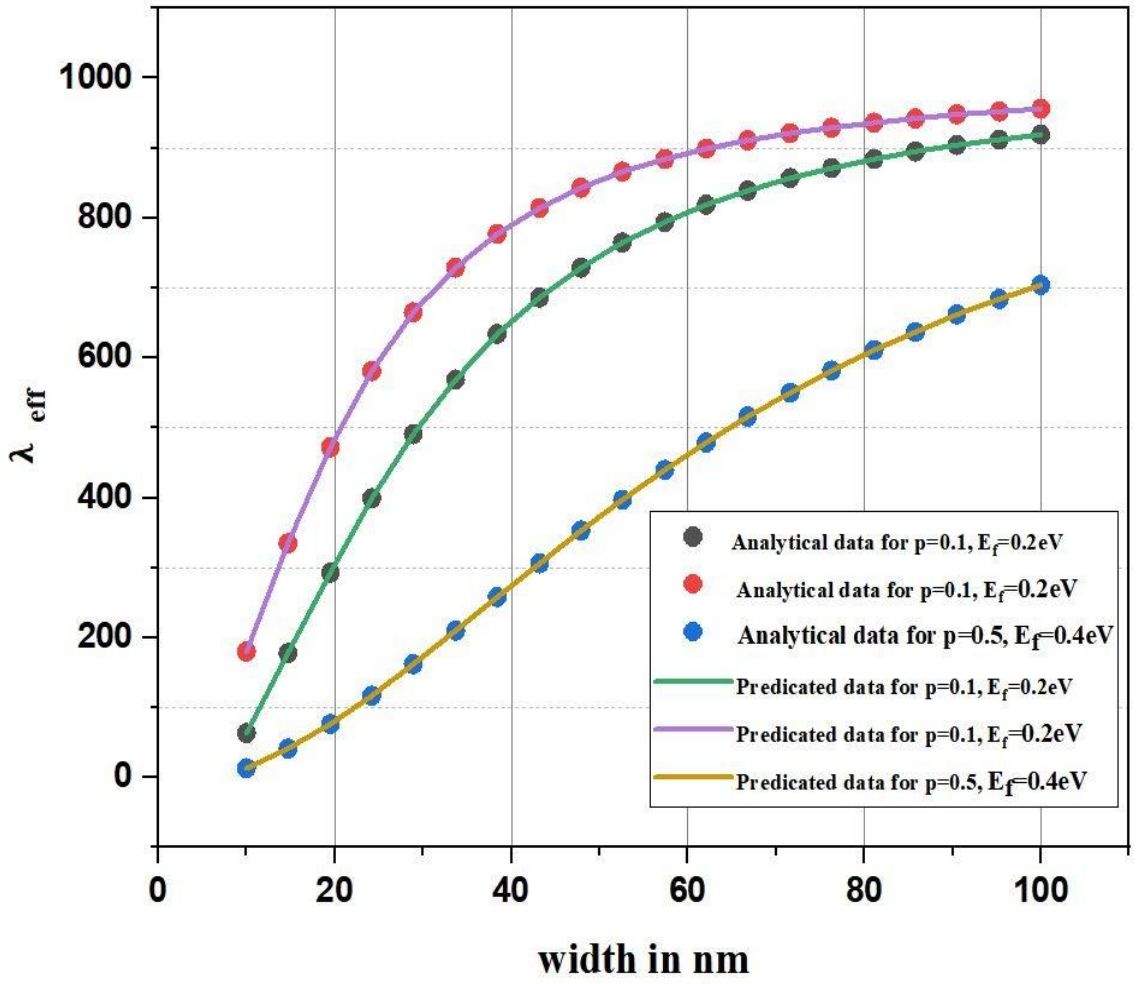
Results and Discussion

We created a combination of linear regression and polynomial regression models to predict the value of effective MFP for the nth sub-band $\lambda_{eff,n}$ for the given values of fermi energy(E_f) .The graph shown in Fig 9. shows us that the graph between $\lambda_{eff,n}$ and E_f is hyperbolic in nature.



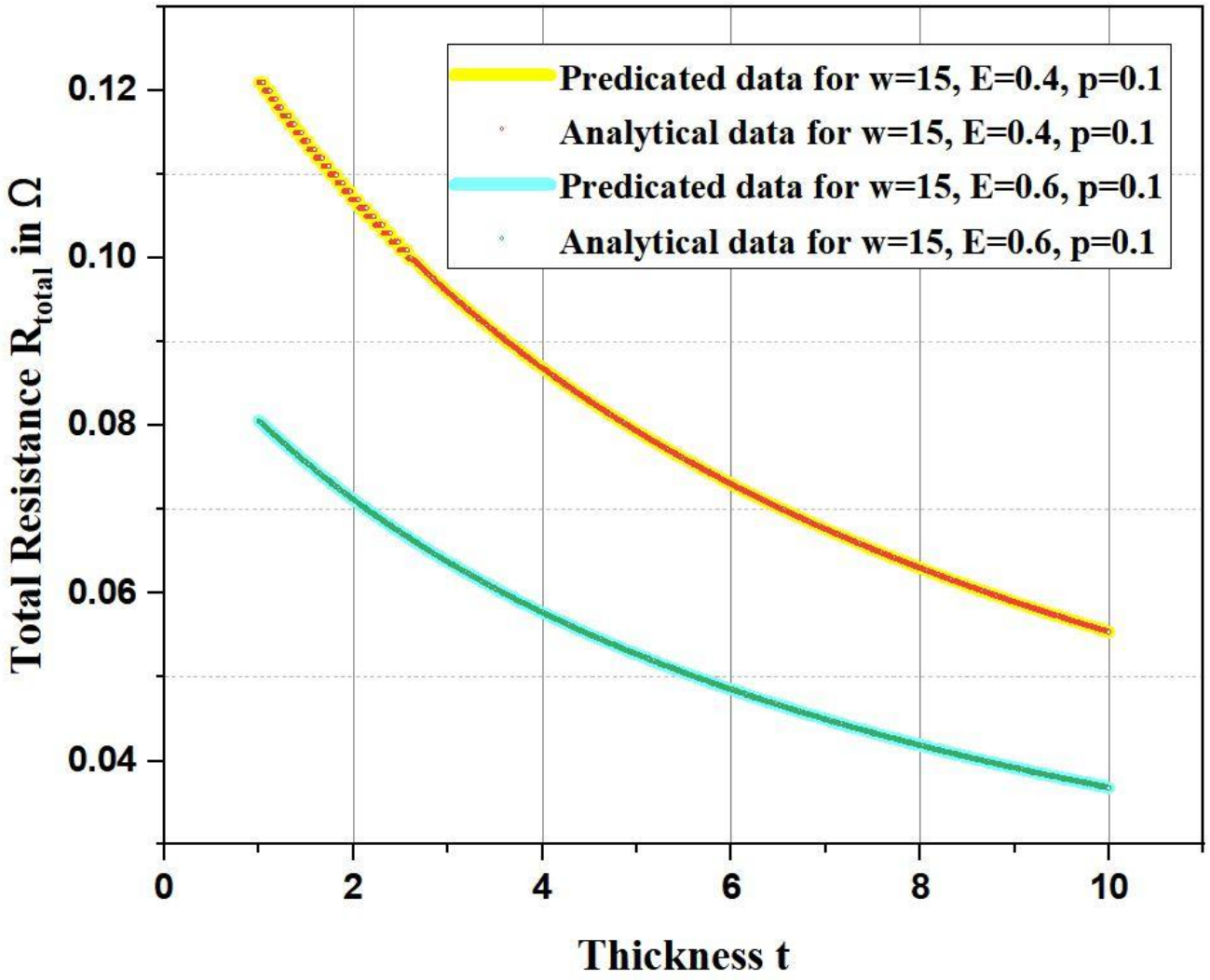
Results and Discussion

We created a combination of linear regression and polynomial regression models to predict the value of effective MFP for the nth sub-band $\lambda_{eff,n}$ for the given values of width in nm(w) .The graph shown in Fig 10. shows us that the $\lambda_{eff,n}$ increases with the decrease in fermi energy and with the decrease of p for the given value of width.



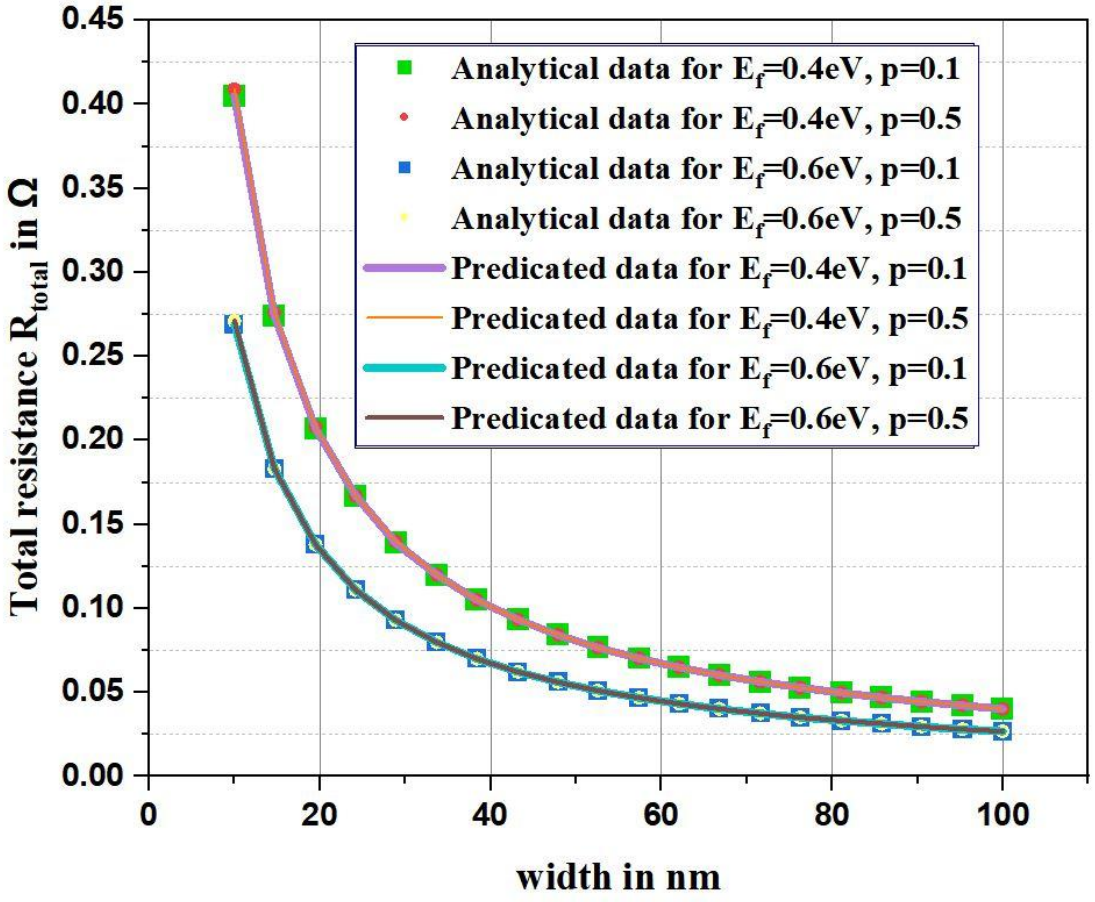
Results and Discussion

We created a combination of linear regression and polynomial regression models to predict the value of total resistance offered by the GNR at the given values of thickness .The graph shown in Fig 11. shows us that the resistance increases with the decrease in fermi energy and for the given value of thickness.



Results and Discussion

We created a combination of linear regression and polynomial regression models to predict the value of total resistance offered by the GNR at the given values of width in nm(w) .The graph shown in Fig 12. shows us that the resistance is not much affected by the values of p and it is inversely proportional to the value of fermi energy.



References

- FU Kai, ZHENG Jie, ZHAO Wen sheng, HU Yue and WANG Gaofeng : Analysis of Transmission Characteristics of Copper/Carbon Nanotube Composite Through-Silicon Via Interconnects* (School of Electronics and Information, Hangzhou Dianzi University, Hangzhou 310018, China)
- Komal Sable, Manodipan Sahoo, Member, IEEE : Electrical and Thermal Analysis of Cu-CNT Composite TSV and GNR Interconnects. Dept. of Electronics and Communication Engineering Dhanbad IIT (ISM), Dhanbad Jharkhand, India.
- . H. Kim, K. Athikulwongse and S. K. Lim, "Study of Through-SiliconVia Impact on the 3-D Stacked IC Layout," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 21, no. 5, pp. 862-874, May 2013.
- J. Jeong, S. Jang, W. Choi, Y. Kim and K. Chun, "Thermal structure design for enhanced heat spreading in 3D ICs," IEEE 2013 Tencon - Spring, Sydney, NSW, 2013, pp. 544-547.
- W. Steinhogl, G. Schindler, G. Steinlesberger, M. Traving, and M. Engelhardt, “Comprehensive study of the resistivity of copper wires with lateral dimensions of 100 nm and smaller,“ Journal of Applied Physics, vol. 97, no. 2, 2005.
- C. Xu, H. Li and K. Banerjee, "Modeling, Analysis, and Design of Graphene Nano-Ribbon Interconnects," in IEEE Transactions on Electron Devices, vol. 56, no. 8, pp. 1567-1578, Aug. 2009.
- D. Jiang, W. Mu, S. Chen, Y. Fu, K. Jeppson and J. Liu, "Vertically Stacked Carbon Nanotube-Based Interconnects for Through Silicon Via Application," in IEEE Electron Device Letters, vol. 36, no. 5, pp. 499- 501, May 2015.
- J. Lee et al., "Understanding Electromigration in Cu-CNT Composite Interconnects: A Multiscale Electrothermal Simulation Study," in IEEE Transactions on Electron Devices, vol. 65, no. 9, pp. 3884-3892, Sept. 2018.
- C. Subramaniam, T. Yamada, K. Kobashi, A. Sekiguchi, D. N. Futaba, M. Yumura, and K. Hata, ”One hundred fold increase in current carrying capacity in a carbon nanotubecopper composite,” Nature Communications, vol. 4, no. 2202, Jul. 2013.
- R. M. Sundaram, A.Sekiguchi, M.Sekiya, T. Yamada and K.Hata, “Copper/carbon nanotube composites: Research trends and outlook,” Royal Society Open Science, vol. 5, no. 11, Nov. 2018
- J. Heremans, J.-P. Issi, I. Zabala-Martinez, M. Shayegan, and M. Dresselhaus, “Thermoelectric properties of a dilute graphite donor intercalation compound,” Physics LettersA, vol. 84, no. 7, pp. 387–389, Aug. 1981.
- J. Park, and V.Prakasha, “Thermal transport in 3D pillared SWCNT– graphene nanostructures,” Journal of Materials Research, vol. 28, no. 7, Apr 14, 2013.
-] W. M. Chanu, V. Prasad and D. Das, "Performance analysis of temperature dependent GNR interconnect," 2016 20th International Symposium on VLSI Design and Test (VDAT), Guwahati, 2016, pp. 1-5.