Numerical Project - FORTRAN

Building of a multidimensional potential energy surface applied to the gas/surface reactivity

Neha Binish

Supervisor: Ludovic Martin-Gondre

Master of Computational Physics
Physics department
Université Bourgogne Franche-Comté

Contents

I PROGRESS OF THE PROJECT

Version: Alpha version

End of Development Date: 5th March, 2021

Programming Language: FORTRAN90

Format: Source code

II LIST OF FILES

In order to increase readability, the project is made of several files. The curve fitting and the plots are made using python3, whereas the development of the LEPS potential energy surfaces is done using FORTRAN90.

`fit.py`: Non - linear curve fitting in Python3

`main.f90` (main program) : Building of the LEPS potentials

`sub.f90`: Modules required to generate 3D-PES scripts and 1-D cuts

`savearray.py`: Python Program to read text files and save as arrays.

`3dplot.py`: Plot of the 3-D-PES script

`error.f90` : Program for Error Analysis

`min.f90` : Program to find the Global Minimum

III FUNCTIONAL SPECIFICATIONS

III.1 Goal of the Program

The goal of this project is to build multidimensional potential energy surface to describe N-N_2 interacting with the tungsten surfaces, specifically for the crystallographic plane W(100).

The potential energy surfaces are build using an analytical model called the LEPS (London Eyring Polanyi Sato) model. Initially, the required potential energies are extracted from the numerical model called the CRP model that has been provided for the N_2/W(100) system. These potential energies are then used to carry out a curve fitting to obtain the various parameters of the LEPS model.

The CRP-PES also behaves as a reference model that serves to compare the quality of the LEPS-PES generated. The attained version of the program builds a 3D-PES for the N/W(100) atom-surface

III.2 Input Data

This program uses several piece of data that can be set by the user to extract the required potential. The potential energies extracted are saved as text file in the folder outputs in the format:

<symmetry-name><crp/leps>.txt

All the files required for the execution of the program are saved in the folder `files`. These files can be loaded into the programs for trial runs.

The following parameters are input by the user to extract the required potential at different symmetry sites.

Parameters	Description	Type
	Input data for extracting potential energies from CRP-PES	
xa	Location on the x-axis in units of delta	Double Precision
ya	Location on the y-axis in units of delta	Double Precision
Zmin	Beginning of the location of the grid on z-axis	Double Precision
Zmax	End of the location of the grid on z-axis	Double Precision
Ntot	Total number of potential energies and distances extracted	Integer
	Input data for main program	
delta (δ)	the unit cell parameter	Real - Double Precision
N	Total number of points in the grid	Integer
Xtop	x-axis location of TOP symmetry site (in units of delta)	Real - Double Precision
Ytop	y-axis location of TOP symmetry site (in units of delta)	Real - Double Precision
Xhollow	x-axis location of HOLLOW symmetry site (in units of delta)	Real - Double Precision
Yhollow	y-axis location of HOLLOW symmetry site (in units of delta)	Real - Double Precision
Xbridge	x-axis location of BRIDGE symmetry site (in units of delta)	Real - Double Precision
Ybridge	y-axis location of BRIDGE symmetry site (in units of delta)	Real - Double Precision
Dtop	Morse potential parameter - D for high symmetry site - TOP	Real - Double Precision
Dhollow	Morse potential parameter - D for high symmetry site - HOLLOW	Real - Double Precision
Dbridge	Morse potential parameter - D for high symmetry site - BRIDGE	Real - Double Precision
alphatop	Morse potential parameter - α for high symmetry site - TOP	Real - Double Precision
alphahollow	Morse potential parameter - α for high symmetry site - HOLLOW	Real - Double Precision
alphabridge	Morse potential parameter - α for high symmetry site - BRIDGE	Real - Double Precision
reqtop	Morse potential parameter - r^{eq} for high symmetry site - TOP	Real - Double Precision
reqhollow	Morse potential parameter - r^{eq} for high symmetry site - HOLLOW	Real - Double Precision
reqbridge	Morse potential parameter - r^{eq} for high symmetry site - BRIDGE	Real - Double Precision

Table 1: Input data

III.3 Output Data

In the alpha version, the program outputs several plots of 1-D cuts for different symmetry sites. These plots are saved in the folder figures in the following format:

<symmetry-name>.png

<1d/3d><symmetry-name>.png

The potential energies extracted from CRP-PES and LEPS-PES, are saved as text files in the folder outputs and later these text files are also used as inputs in the given format:

<p/z><crp/pes>_<symmetry-name>.txt

The figures generated are stored in the folder `figures`.

In addition to this, the program also computes the following parameters:

Parameters	Description	Type
Coeff_D	Fourier coefficients of the Morse potential parameter - D	Real - Double Precision - size(3,1)
Coeff_alpha	Fourier coefficients of the Morse potential parameter - α	Real - Double Precision - size(3,1)
Coeff_req	Fourier coefficients of the Morse potential parameter - r^{eq}	Real - Double Precision - size(3,1)
M_error	Mean absolute error	Real - Double Precision
REL_error	Relative error	Real - Double Precision
CRP_min	Global minimum of the CRP-PES	Real - Double Precision
LEPS_min	Global minimum of the LEPS-PES	Real - Double Precision

Table 2: Output data

III.4 Specific Conditions of Use

- The README file contain instructions on how to generate potential energies from the CRP-PES scripts (energies are in eV and distances are in Å). However, the user must previously generate the executable file `textat.exe`, without which the script fails to run
- On extracting the CRP-PES potentials and distances at different positions and symmetry, curve fitting was attempted. It was found that when the values from zmin = 0 was included, curve fitting using python polyfit was not possible as the function was unable to find the right parameters for the fit due to the high fluctuation of the potential around zmin = 0, when it tends to infinity.
- Hence, for a good for using polyfit function in python 3, the first few values of the potential was cut off for specific symmetry sites and the potentials are extracted from distances zmin = 0.5.
- Moreover, for error optimization, the user is required to choose the right range of values on the z-axis over which the potential energies are extracted.
- The user is also required to have all the modules pre-loaded and compiled.

- In order to run a trial with the existing text files, it is necessary to include the path of these files and the user also has to take care not to rewrite the files or false results could be produced.

IV INTERNAL FUNCTIONING OF THE PROGRAM

IV.1 Physical Model

IV.1.1 CRP-PES

To interpret and compare the LEPS potential, it is necessary to extract the potentials energies from the CRP-PES for various positions of N onto W(100) surface.

We refer to the README file in order to generate potential energies from the CRP-PES scripts provided. The output data consisting of potential energies (in eV) and the corresponding distances (in Å) are extracted for different symmetry sites. For this particular project - the data was extracted for the following points -

Symmetry Sites	xa (in units of delta)	ya (in units of delta)	zmin	zmax	Ntot
High Symmetry Site - top	0	0	0.5	10.0	200
High Symmetry Site - bridge1	0.5	0	0.5	10.0	300
High Symmetry Site - bridge2	0	0.5	0.5	10.0	300
High Symmetry Site - hollow	0.5	0.5	0.0	6.0	200
Low Symmetry Site - top-hollow	0.25	0.25	0.0	10.0	200
Low Symmetry Site - top-bridge	0.25	0	0.0	10.0	200
Low Symmetry Site - bridge-hollow	0.5	0.25	0.0	10.0	200

Table 3: Data Extraced from CRP-PES

IV.1.2 LEPS-PES

In the alpha version of this project, we develop a simplified 3-dimensional potential to describe the N/W(100) interactions. In order to do so, we perform a non-linear fitting on the three high symmetry sites - top, bridge and hollow, using the potential energies extracted from the CRP-PES scripts provided. This curve fitting is carried out in Python 3. On plotting, the extracted potential energies, we deduce that the atom-surface interactions in conventional LEPS potentials, can be modelled with a Morse function -

$$V^{3D}(r_{at}) = \mathrm{D}\Big[\exp\big(-2\alpha(Z_{at} - r^{eq})\big) - 2\exp\big(-\alpha(Z_{at} - r^{eq})\big)\Big]$$

From the non-linear fitting, we obtain the three parameters of this morse function; D - is the well depth related to the molecule dissociation energy, α - defining the range of the potential and

the r^{eq} - is the equilibrium distance between the atom and the solid surface. Once, we have the Morse parameters, we can define the Fourier expansion for the (100) symmetry of the bcc crystal as -

$$\begin{aligned}\text{Four}(X_{at}, Y_{at}) = P_0 + P_1\Big(cos\frac{2\pi X_{at}}{\delta} + cos\frac{2\pi Y_{at}}{\delta}\Big) + P_2\Big(cos\frac{2\pi(X_{at}+Y_{at})}{\delta} + cos\frac{2\pi(X_{at}-Y_{at})}{\delta}\Big) \\ + P_3(cos\frac{4\pi X_{at}}{\delta} + cos\frac{4\pi Y_{at}}{\delta}) \\ + P_4(cos\frac{2\pi(2X_{at}+Y_{at})}{\delta} + cos\frac{2\pi(X_{at}+2Y_{at})}{\delta} \\ + cos\frac{2\pi(2X_{at}-Y_{at})}{\delta} + cos\frac{2\pi(X_{at}-2Y_{at})}{\delta}) \\ + P_5(cos\frac{4\pi(X_{at}+Y_{at})}{\delta} + cos\frac{4\pi(X_{at}-Y_{at})}{\delta})\end{aligned}$$

Here, δ is the unit cell parameter calculated with ab-initio methods (here with DFT : Density Functional Theory) is equal to 3.17 Å and the P_i are the Fourier coefficients. The values for X_{at} and Y_{at} can be chosen from the Table:3 according to the choice of symmetry sites.

A periodic dependence in (X_{at}, Y_{at}) is introduced in the three Morse Potential Parameters with the Fourier development due to the surface corrugation. In the alpha version, we have chosen the three high symmetry sites - top, bridge and hollow where the co-ordinates (X_{at}, Y_{at}) are well defined. Thus we have a second order Fourier expansion with the P_0, P_1 and P_2 coefficients for the three Morse Parameters given below -

$$\text{D}(X_{at}, Y_{at}) = P_0 + P_1\Big(cos\frac{2\pi X_{at}}{\delta} + cos\frac{2\pi Y_{at}}{\delta}\Big) + P_2\Big(cos\frac{2\pi(X_{at}+Y_{at})}{\delta} + cos\frac{2\pi(X_{at}-Y_{at})}{\delta}\Big)$$

$$\alpha(X_{at}, Y_{at}) = P_0 + P_1\Big(cos\frac{2\pi X_{at}}{\delta} + cos\frac{2\pi Y_{at}}{\delta}\Big) + P_2\Big(cos\frac{2\pi(X_{at}+Y_{at})}{\delta} + cos\frac{2\pi(X_{at}-Y_{at})}{\delta}\Big)$$

$$r^{eq}(X_{at}, Y_{at}) = P_0 + P_1\Big(cos\frac{2\pi X_{at}}{\delta} + cos\frac{2\pi Y_{at}}{\delta}\Big) + P_2\Big(cos\frac{2\pi(X_{at}+Y_{at})}{\delta} + cos\frac{2\pi(X_{at}-Y_{at})}{\delta}\Big)$$

Thus, using the Morse potential parameters that have been obtained from the non-linear curve fittings, we can solve a system of linear equations defined according to the values of (X_{at}, Y_{at}) coordinates of the three high symmetry sites - top, hollow and bridge to obtain the P_i coefficients. Then, we have the following system of equations -

$$\begin{bmatrix} D_{top} \\ D_{bridge} \\ D_{hollow} \end{bmatrix} = \begin{bmatrix} 1 & \left(cos\frac{2\pi X^{top}}{\delta} + cos\frac{2\pi Y^{top}}{\delta}\right) & \left(cos\frac{2\pi (X^{top}+Y^{top})}{\delta} + cos\frac{2\pi (X^{top}-Y^{top})}{\delta}\right) \\ 1 & \left(cos\frac{2\pi X^{bridge}}{\delta} + cos\frac{2\pi Y^{bridge}}{\delta}\right) & \left(cos\frac{2\pi (X^{bridge}+Y^{bridge})}{\delta} + cos\frac{2\pi (X^{bridge}-Y^{bridge})}{\delta}\right) \\ 1 & \left(cos\frac{2\pi X^{hollow}}{\delta} + cos\frac{2\pi Y^{hollow}}{\delta}\right) & \left(cos\frac{2\pi (X^{hollow}+Y^{hollow})}{\delta} + cos\frac{2\pi (X^{hollow}-Y^{hollow})}{\delta}\right) \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_1 \end{bmatrix}$$

$$\begin{bmatrix} \alpha_{top} \\ \alpha_{bridge} \\ \alpha_{hollow} \end{bmatrix} = \begin{bmatrix} 1 & \left(cos\frac{2\pi X^{top}}{\delta} + cos\frac{2\pi Y^{top}}{\delta}\right) & \left(cos\frac{2\pi (X^{top}+Y^{top})}{\delta} + cos\frac{2\pi (X^{top}-Y^{top})}{\delta}\right) \\ 1 & \left(cos\frac{2\pi X^{bridge}}{\delta} + cos\frac{2\pi Y^{bridge}}{\delta}\right) & \left(cos\frac{2\pi (X^{bridge}+Y^{bridge})}{\delta} + cos\frac{2\pi (X^{bridge}-Y^{bridge})}{\delta}\right) \\ 1 & \left(cos\frac{2\pi X^{hollow}}{\delta} + cos\frac{2\pi Y^{hollow}}{\delta}\right) & \left(cos\frac{2\pi (X^{hollow}+Y^{hollow})}{\delta} + cos\frac{2\pi (X^{hollow}-Y^{hollow})}{\delta}\right) \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_2 \end{bmatrix}$$

$$\begin{bmatrix} r^{eq}_{top} \\ r^{eq}_{bridge} \\ r^{eq}_{hollow} \end{bmatrix} = \begin{bmatrix} 1 & \left(cos\frac{2\pi X^{top}}{\delta} + cos\frac{2\pi Y^{top}}{\delta}\right) & \left(cos\frac{2\pi (X^{top}+Y^{top})}{\delta} + cos\frac{2\pi (X^{top}-Y^{top})}{\delta}\right) \\ 1 & \left(cos\frac{2\pi X^{bridge}}{\delta} + cos\frac{2\pi Y^{bridge}}{\delta}\right) & \left(cos\frac{2\pi (X^{bridge}+Y^{bridge})}{\delta} + cos\frac{2\pi (X^{bridge}-Y^{bridge})}{\delta}\right) \\ 1 & \left(cos\frac{2\pi X^{hollow}}{\delta} + cos\frac{2\pi Y^{hollow}}{\delta}\right) & \left(cos\frac{2\pi (X^{hollow}+Y^{hollow})}{\delta} + cos\frac{2\pi (X^{hollow}-Y^{hollow})}{\delta}\right) \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_2 \end{bmatrix}$$

Solving the above equations, we get the three coefficients P_0, P_1 and P_2 for each of the Morse Potential parameters. Using these coefficients, we create the 3D-PES script and plot 1D-cuts of the for the three high symmetry sites (top, bridge, hollow) and of three lower symmetry sites (top-bridge, top-hollow, bridge-hollow). The LEPS curves are then compared to the CRP curves.

We also find the global minimum of the CRP-PES and the LEPS-PES calculate an plot 1-D cuts for the for the three high symmetry sites (top, bridge, hollow) and three lower symmetry sites (top-bridge, top-hollow, bridge-hollow). Moreover, we also calculate an estimation of the mean error for every site considered with respect to the CRP potential energies.

IV.2 Constituent Elements

IV.2.1 `fit.py`

This Python 3 file reads the text files containing the potential energies and distances extracted from the CRP-PES scripts and carries out a non-linear fitting on these potentials using the np.polyfit() function from the NumPy library. It returns the three Morse parameters; D, α and r^{eq} for each symmetry site.

IV.2.2 `main.f90`

This is the main program that carries out calculations to build the LEPS potentials for the alpha version of the program. This file makes use of the module **`sub.f90`** for the required subroutines that help generate the 3D - PES scripts and plot the 1D cuts.

IV.2.3 `sub.f90`

This a module file contains the subroutines that are required for the execution of the main program that is used to build the LEPS potentials for the alpha version of the program. This file has several subroutines to ease the execution of the program.

1. A subroutine to create an equally spaced array between two given points.

`subroutine linspace (from, to, array)`			
parameter	**intent - (input/output)**	**description**	**type**
from	input	Starting point for the grid (the first value of the array)	REAL (double precision)
to	input	Ending point for the grid (the last value of the array)	REAL (double precision)
array	output	Equally spaced array made for the given range	REAL (double precision)

Table 4: Subroutine 1 - linspace

2. A subroutine that generates the 3-D PES script for a given point on the x-axis and a range of values on the y and the z-axis.

`subroutine PES3D (X, Y, Z, Dcoeff, Acoeff, Rcoeff, N)`			
parameter	**intent - (input/output)**	**description**	**type**
X	input	The fixed point on the x-axis over which the potential energy surfaces are evaluated	REAL (double precision)
Y	input	Array of values on the y-axis over which the potential energy surfaces are evaluated	REAL (double precision) - Array of dimension N
Z	input	Array of values on the z-axis over which the potential energy surfaces are evaluated	REAL (double precision) - Array of dimension N
Dcoeff	input	Fourier Coefficients of the Morse parameter D	REAL - double precision - Dimension(3,1)
Acoeff	input	Fourier Coefficients of the Morse parameter α	REAL - double precision - Dimension(3,1)
Rcoeff	input	Fourier Coefficients of the Morse parameter r^{eq}	REAL - double precision - Dimension(3,1)
N	input	Total number of points in the grid.	INTEGER
y_arr3d.txt	output	Range of values on the y-axis over which the potentials are generated	text file
z_arr3d.txt	output	Range of values on the z-axis over which the potentials are generated	text file
potential3d.txt	output	Two-dimensional potential evaluated over a range of values at a fixed point on x-axis	text file

Table 5: Subroutine 2 - PES3D

3. A subroutine to generate 1-D cuts for he symmetry sites.

`subroutine plot1D(x, y, N, Dcoeff, Acoeff, Rcoeff, Z)`			
parameter	**intent - (input/output)**	**description**	**type**
x	input	x co-ordinate of the selected symmetry site	REAL (double precision)
y	input	y co-ordinate of the selected symmetry site	REAL (double precision)
N	input	Total number of points in the grid.	INTEGER
Dcoeff	input	Fourier Coefficients of the Morse parameter D	REAL - double precision - Dimension(3,1)
Acoeff	input	Fourier Coefficients of the Morse parameter α	REAL - double precision - Dimension(3,1)
Rcoeff	input	Fourier Coefficients of the Morse parameter r^{eq}	REAL - double precision - Dimension(3,1)
Z	input	Array of values on the z-axis over which the potential energy surfaces are evaluated	REAL (double precision) - Array of dimension N
potential.txt	output	Text file containing the potentials evaluated	text file
distances.txt	output	Text file containing the range at which the potentials are evaluated	text file

Table 6: Subroutine 3 - plot1D

IV.2.4 `min.f90`

This is the program written to find the global minimum of all the LEPS-PES and the CRP-PES. All the files generated in the previous programs are read and loaded into this program. The minimum value of these files are found and compared to find the least minimum value. This program makes use of the following subroutine -

1. A subroutine that reads files and loads it's contents into an allocated array in the program. It also returns the total number of lines in the files.

`subroutine read_file(filename, value, count)`			
parameter	**intent - (input/output)**	**description**	**type**
filename	input	Input text file that has to be read.	CHARACTER
value	output	The values that are read from the file and loaded into the program	REAL - (kind = 8) - allocatable array
count	output	Total number of lines in the file	REAL - (kind = 8)

Table 7: Subroutine 1 - read_file

IV.2.5 `error.f90`

This program calculates an estimation of the mean error and mean deviation for the symmetry sites - top, bridge, hollow, top-bridge, top-hollow and bridge-hollow. It makes use of the following formula for the calculation of error -

$$\text{Absolute Error} = \textbf{mean}\left(\left| \text{Potential LEPS(I) - Potential CRPS(I)} \right| \right)$$

$$\text{Relative Error} = \textbf{mean}\left(\left| \frac{\text{Potential CRP (I)} - \text{Potential LEPS (I)}}{\text{Potential CRP (I)}} \right| \right)$$

$$\text{Relative Percent Error} = \textbf{mean}\left(\left| \frac{\text{Potential CRP (I)} - \text{Potential LEPS (I)}}{\text{Potential CRP (I)}} \right| \right) \times 100\%$$

1. A subroutine that reads files and loads it's contents into an allocated array in the program. It also returns the total number of lines in the files.

`subroutine read_file(filename, value, count)`			
parameter	**intent - (input/output)**	**description**	**type**
filename	input	Input text file that has to be read.	CHARACTER
value	output	The values that are read from the file and loaded into the program	REAL - (kind = 8) - allocatable array
count	output	Total number of lines in the file	REAL - (kind = 8)

Table 8: Subroutine 1 - read_file

IV.2.6 `CRP_N_plot.py`

In this program, we plot the 1-D cuts of the initially extracted CRP-PES potential energy surfaces against the distances for each symmetry sites evaluated.

IV.2.7 `plot.py`

This program file is created in Python 3. It has a function that reads and loads the text files generated in the main program. These loaded text files, provide the extracted potentials and the respective distances for each symmetry site evaluated in the main program for the LEPS potentials that were build. Finally, this program plots the 1-D cuts of the LEPS curves that are generated.

IV.2.8 `3dplot.py`

This program file is also created in Python 3 and contains a function that reads the outputs of the 3D PES-Script. It loads the two-dimensional Potential and the z as well as y-axis arrays. This program then plots a three dimensional graph for the given values.

V FUNCTIONAL DIAGRAM

The schematic flow of the programs are explained below using the Information diagram. The steps followed for the execution of this program is shown in the Functional specification.

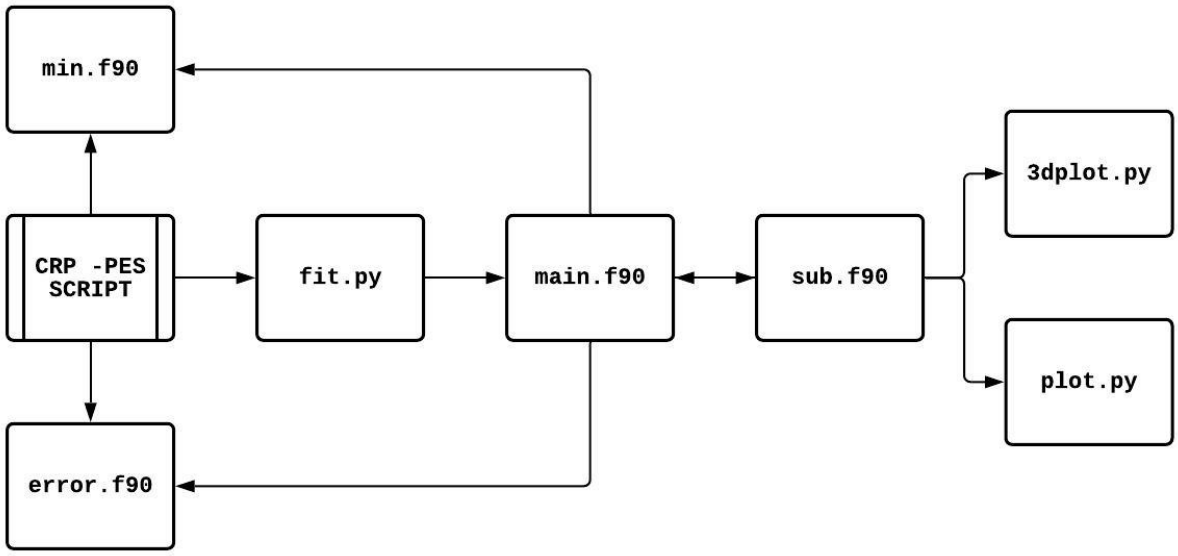


Figure 1: Information Diagram on flow of the program

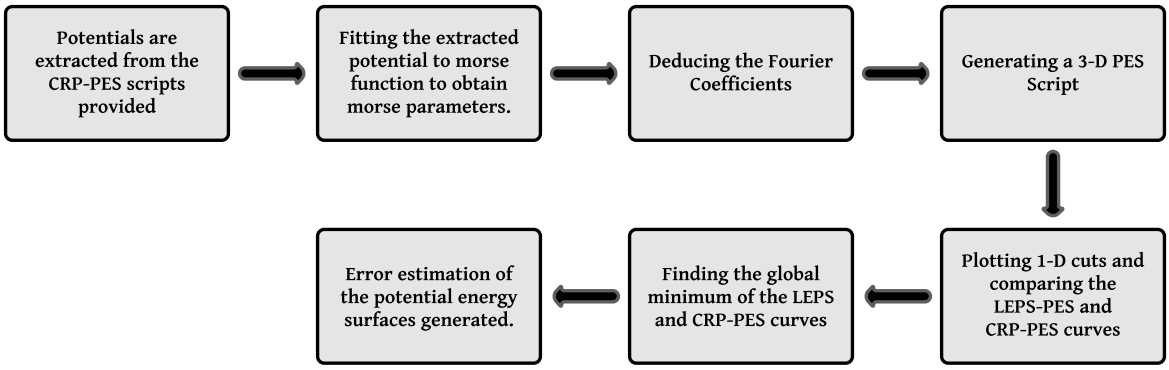


Figure 2: Functional Diagram of the program

VI RELIABILITY OF THE PROGRAM - QUALITY APPROACH

---

1. The LEPS potentials build by the program where compared to the CRP-PES obtained from the scripts that were provided. After the analysis of the results such as comparison of the LEPS and CRP-PES curves. It can be conculded that the program is reliable and can be used to produce LEPS potential energies that follow the trend of the CRP potential energies.
2. For the curve fitting of the extracted CRP-PES potential energies, it is important to have good initial parameters. These are however guesses and the curve fitting errors, if any, could be related to this problem.
3. Moreover, we have also carried out a detailed error analysis of the obtained potential energies. The absolute error was found to be of an tolerable range. Whereas the relative error is dependent on the range of values on the z-axis over which the potential energies are extracted. For lesser error factors, the user should choose the range of z-axis closer to the potential well as the error factor increases as we move away from this region. This error optimization technique produces LEPS potential energies that are comparable to the CRP-PES potential energies
4. The CPU run time was found to be very less in the order of milliseconds for the execution of the programs. Only the loading of large text files may cause a delay in the run time but this program was executed for total number of potential values in the range of N = 100-400. This execution did not produce any large text files and kept the CPU run time very low. The largest text file was that of the two dimensional potential energies generated by the 3-D PES script and it was found to be in the order of 2 Megabytes.
5. For memory optimization, most arrays that have been defined are allocatable arrays that are not pre-defines memory spaces. Depending on the size of the text files, we allocate the memory to the arrays and after usage deallocate them
6. It is also possible to evaluate the CRP-PES for a larger number of potential energies. This would help in the optimization of the nonlinear curve fit that is produced.

7. Following this, the LEPS potential energies that are build will also be at a large number of points and the potential energies evaluated at closer intervals would give lesser chances for the occurrence of errors.
8. All the data that has been used for the execution of the program and and all the results obtained after the execution has been stored in the folder `files`. These data can be extracted and used by the user to check for the reliability of this program.

VII OUTCOME EXAMPLES

VII.1 CRP-PES

The initial step of this project was to extract the potential energies from the provided CRP-PES scripts for various positions of N onto W(100) surfaces. The following 1-D cuts of the potential energy as a function of the N/W distance were obtained for the three high-symmetry sites; coordinates can be referred to in Table: 3. These plots are made for a set of 300 points over the range of zmin = -2 Å to zmax = 6 Å.

It can be noticed that the CRP-PES curves in figure: 3 are like that of the Morse potential curves and hence we use the Morse function to model these curves in a non-linear curve fit. The 1-D cuts are plotted for the Bridge-1 and Bridge-2 coincide with each other and hence this can be chosen as a single high symmetry site - Bridge.

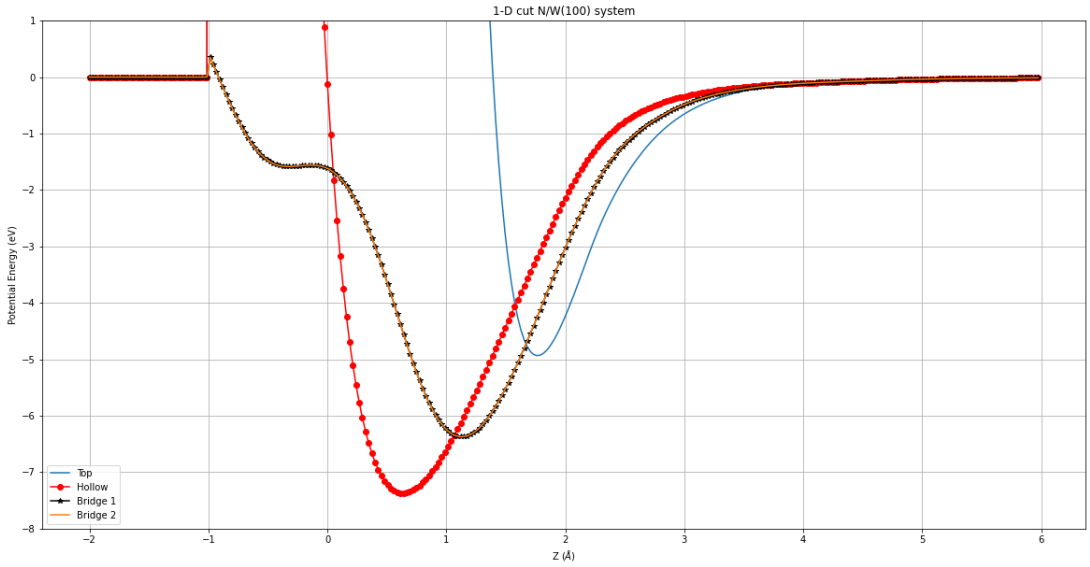


Figure 3: 1-D cut: CRP-PES

VII.2 Non-linear Curve Fit

In order to deduce the Fourier Coefficients, we need to obtain the Morse Potential Parameters. These parameters are obtained by performing a non-linear curve fit onto the CRP-PES curves with the Morse potential model.

The initial parameters are set as:

1. Initial Parameters for Top Symmetry : [-8.0, 2.0, 3.0]
2. Initial Parameters for Hollow Symmetry : [-6.0, 1.0, 0.1]
3. Initial Parameters for Bridge Symmetry : [-6.0, 1.0, 1.0]

It was found that, when the range of the distances in the z-axis was takes to be negative, the Bridge and the Hollow symmetry showed unusual behaviour. It also made the modelling use the Morse function very difficult and hence these values were then excluded during the fitting procedure. Moreover, the behaviour of the high symmetry sites were such that they quickly approached extremely high potential energy values when the distance on the z-axis approached zero. These values once again, prevented np.polyfit from providing good fits for the plots. Hence, for the fitting procedure the range of distances in the z-axis was taken as zmin = 0.5 Å to zmax = 6-10 Å. In this range we have removed the extremely fluctuating values of the potential to obtain a good fit.

We get the following values for the Morse Potential Parameters:

1. Parameters for hollow symmetry are: [7.60346595, 1.2495198, 0.56501317]
2. Parameters for bridge symmetry are: [6.55379066, 1.30169922, 0.95376732]
3. Parameters for top symmetry are : [4.9984788, 1.71646688, 1.83275503]

The curve fit for each of the high symmetry sites - Top, Hollow and Bridge are given in the figures: 4, 5 and 6. We can conclude that the parameters obtained are good parameters by looking at the below fit. We see a fairly good coherence between the theoretical CRP potentials and the non-linear curves that have been fit.

The NumPy library makes use of numpy.polyfit() that carries out a Least-squares polynomial fit. However, this procedure doesn't necessarily have to be done using the pre-defined NumPy function. We could also manually write a FORTRAN or a Python code for the Least-Squares fit.

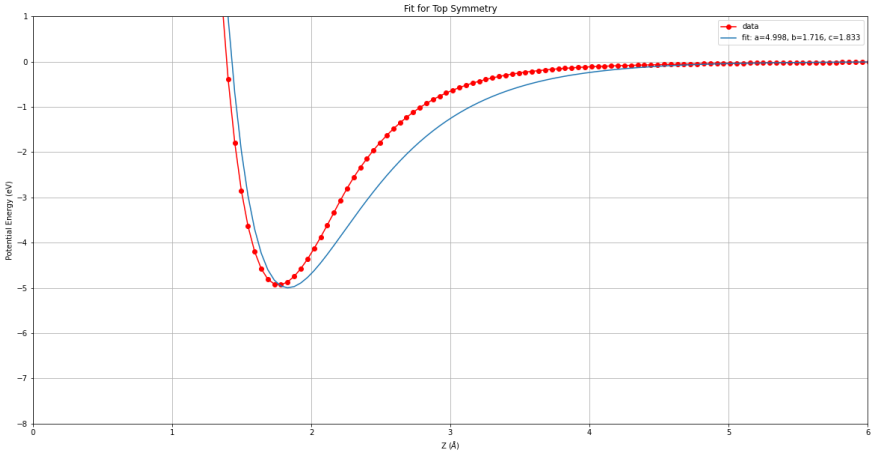


Figure 4: Non-linear curve fit for the High Symmetry Site - Top

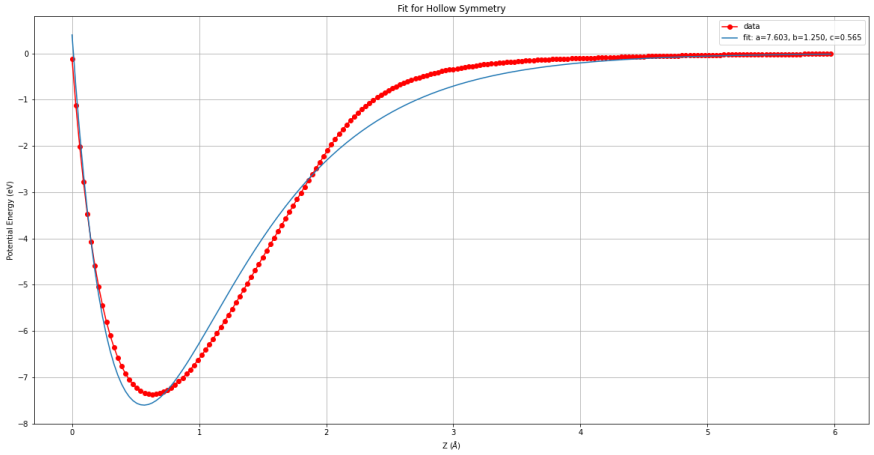


Figure 5: Non-linear curve fit for the High Symmetry Site - Hollow

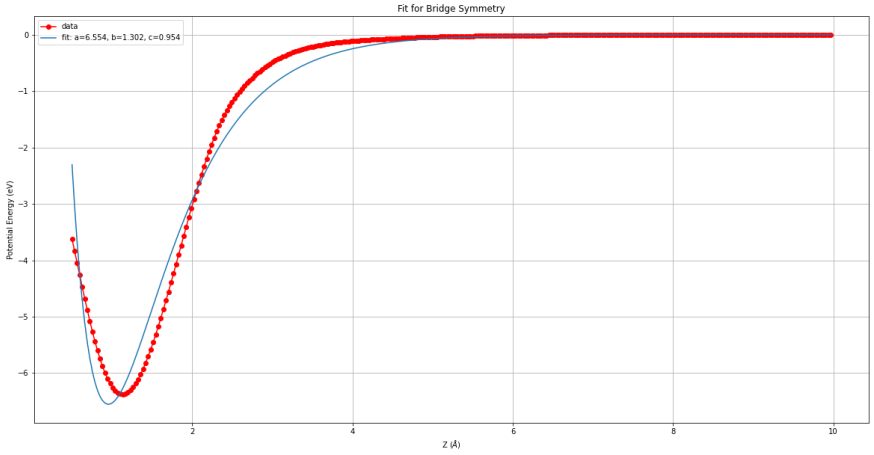


Figure 6: Non-linear curve fit for the High Symmetry Site - Bridge

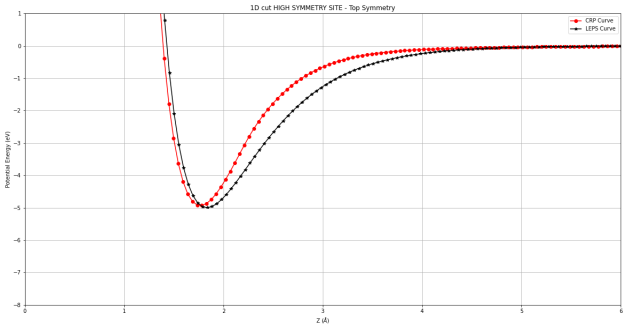
VII.3 Fourier Coefficients and Comparison of LEPS and CRP-PES curves

Using the parameters obtained from the previous section, we were able to calculate the required Fourier Coefficients for each Morse Potential Parameter:

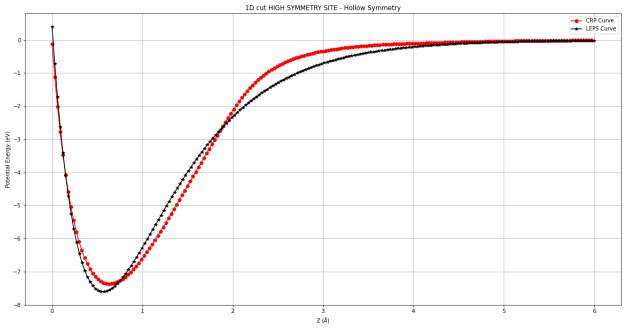
Morse Potential Parameters	**First Fourier Coefficient** $-P_0$	**Second Fourier Coefficient** - P_1	**Third Fourier Coefficient** - P_3
D	8.9587	-1.1614	-0.8187
α	1.3979	- 0.4278	0.5871
r^{eq}	0.2568	- 5.7649E-003	0.7938

Table 9: Fourier Coefficients for the Morse Potential Parameters

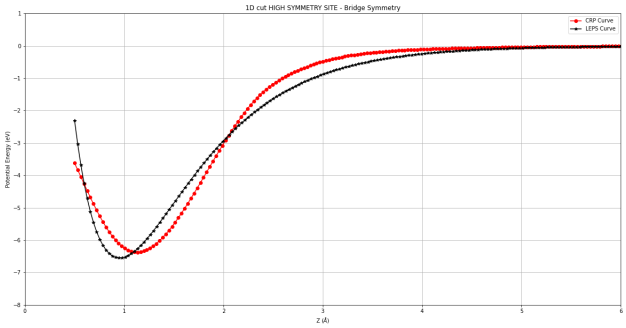
From these Fourier Coefficients, we obtain the following plots for the three high symmetry sites (top, bridge, hollow) and for the three lower symmetry sites (top-bridge, top-hollow, bridge-hollow). We also plot the comparison between the three HIGH symmetry sites in figure: 10a and the three LOW symmetry sites in the figure: 10b. Finally, we also provide a comparison of all the symmetry sites in the figure: 11.



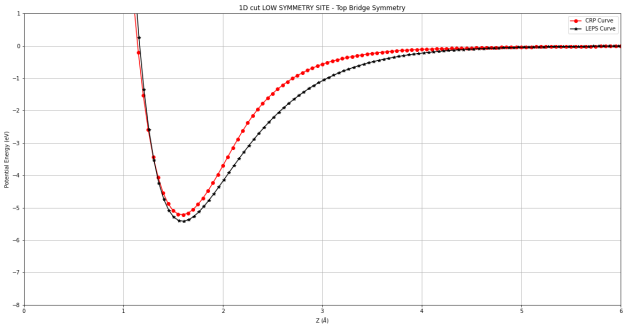
(a) 1-D cut LEPS curve - Top



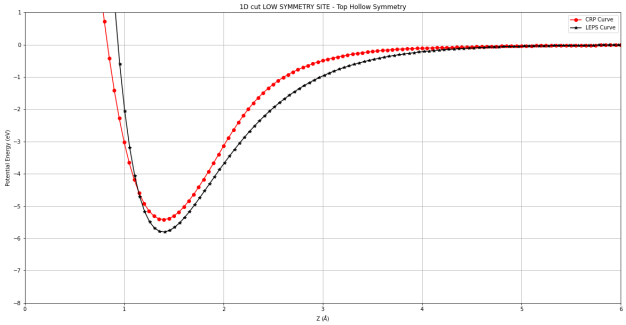
(b) 1-D cut LEPS curve - Hollow



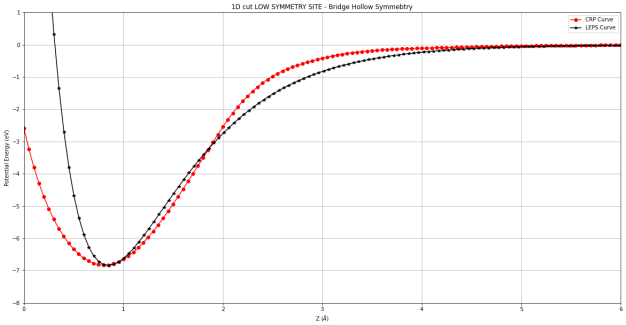
(a) 1-D cut LEPS curve - Bridge



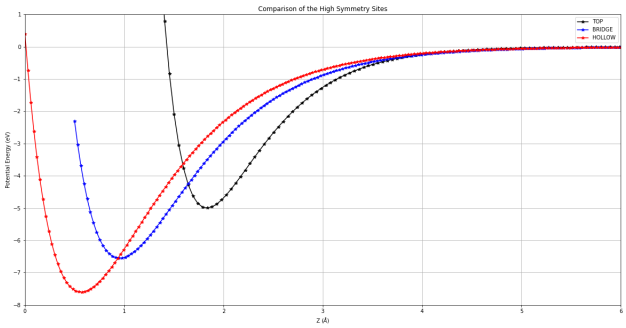
(b) 1-D cut LEPS curve - Top Bridge



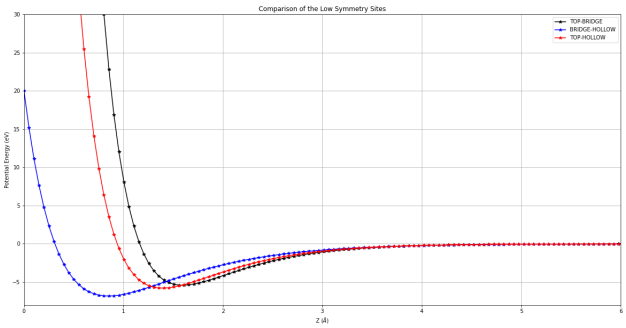
(a) 1-D cut LEPS curve - Top Hollow



(b) 1-D cut LEPS curve - Bridge Hollow



(a) 1-D cut LEPS curve - High symmetry sites



(b) 1-D cut LEPS curve -Low symmetry sites

Figure 10: Comparison of LEPS and CRP-PES curves

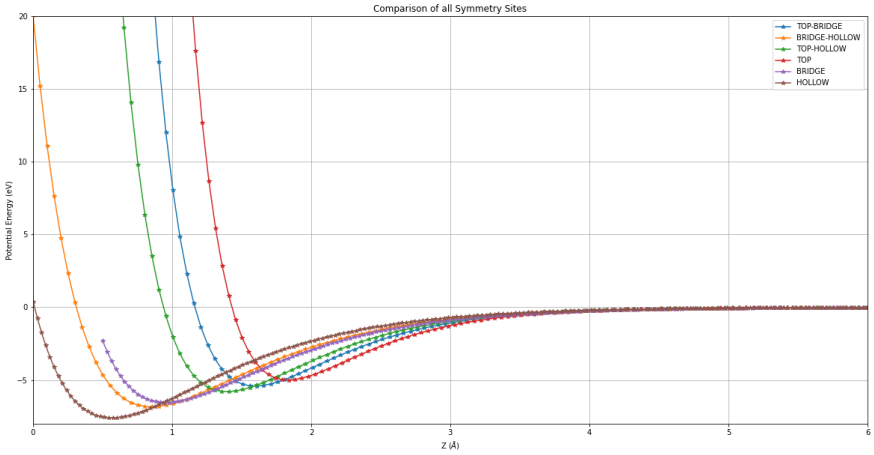


Figure 11: Comparison of all the symmetry sites

VII.4 Testing of the 3-D PES Script

The 3D PES script generated by making the following three dimensional plots from the two dimensional potential plotted against the range of values on the y-axis and z-axis for a fixed x-value.

In this program, the 3-D scripts are done for a fixed point on the x-axis and a range of values on the y and z-axis. However with slight modification to the program, it is possible to generate

these scripts for fixed points on y and z-axis and evaluating the potential energies over the other two axis.

The plot generated in figure: 12 shows the behaviour of a Morse function with a depth D around -6 eV and the potential increasing as z reached zero. Hence, the nature of the 3D-PES script has been tested and it is believed to be accurate from the obtained results.

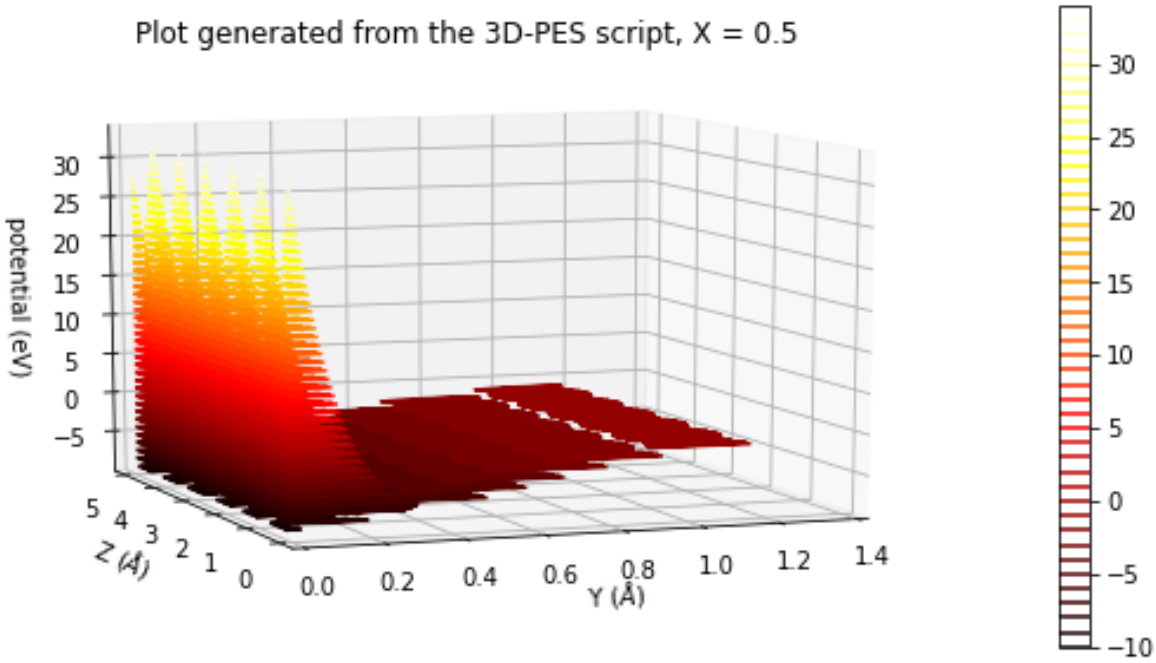


Figure 12: Plot generated from the 3D-PES script for X=0.5Å

VII.5 Global Minimum

On the comparison of all the minimum values of the LEPS and CRP-PES curves, we were able to find the Global minimum for these two curves for the atom/surface by associating the energy barrier to reach this energy minimum. We have the following results

The Global minimum for the **CRP** - Potential Energy Surfaces: -7.37103987 eV

The Global minimum for the **LEPS** - Potential Energy Surfaces: -7.60317278 eV

VII.6 Error analysis

Due to the nature of the Morse function, it is not possible to directly calculate the mean error, instead we find the deviation of each LEPS potential energy from that of the CRP potential energies. The error is calculated for each point after which a mean error is calculated. We get the following results -

Mean absolute error for High Symmetry Site: **Hollow** - 0.212

Mean relative error for High Symmetry Site: **Hollow** - 53.927%

Mean absolute error for High Symmetry Site: **Bridge** - 0.148

Mean relative error for High Symmetry Site: **Bridge** - 235.399%

Mean absolute error for High Symmetry Site: **Top** - 0.324

Mean relative error for High Symmetry Site: **Top** - 78.009%

Mean absolute error for Low Symmetry Site: **Top Bridge** - 15.347

Mean relative error for Low Symmetry Site: **Top Bridge** - 115.161%

Mean absolute error for Low Symmetry Site: **Top Hollow** - 7.359

Mean relative error for Low Symmetry Site: **Top Hollow** - 162.949%

Mean absolute error for Low Symmetry Site: **Bridge Hollow** - 0.627

Mean relative error for Low Symmetry Site: **Bridge Hollow** - 254.640%

The error factor received from the Relative error calculation was huge even though the fit produces looked coherent. After several tries to debug the code and thorough analysis of the potential energies studied, it was realised that the huge error factor was result of the long range of distances in the z-axis. As the z axis extended beyond 6-7 Å, it was found that the error between the CRP potential and LEPS potential points increased drastically. However, this is not visible on the plots due to extremely small values of potentials in this region. This hypothesis was rechecked by running an error analysis on the potentials in the range 0.5-7 Å. We obtain the following results:

```
Mean absolute error for High Symmetry Site: Bridge -   0.14808183950796958
Mean relative error for High Symmetry Site: Bridge -    235.39944395341536      %

Mean absolute error for High Symmetry Site: Top -   0.32367000398004042
Mean relative error for High Symmetry Site: Top -    78.009786725824682      %
```

Figure 13: Error analysis for z - 0.5 to 10Å

```
 Mean absolute error for High Symmetry Site: Bridge -   0.21504502864833067
 Mean relative error for High Symmetry Site: Bridge -    59.921847820260183      %

 Mean absolute error for High Symmetry Site: Top -   0.47228316167569323
 Mean relative error for High Symmetry Site: Top -    49.421399485736735      %
```

Figure 14: Error analysis for z - 0.5 to 7Å

On comparison of the two figures:13 and 14 , one can see the drastic decrease in the values of the Relative errors. The relative error for the High symmetry site - bridge drops from 235% to just around 60%. Similarly there is also a drop in the error percentage of the High Symmetry site - Top. This decrease in the error factor was verified for all symmetry sites and hence it can be concluded that the range at which the LEPS potential energies are evaluated, play an important role in the reduction of error.

VIII CONCLUSION

The main objective of this project was to develop a simplified three dimensional potential to describe the N/W(100) interactions. We were able to obtain LEPS potential values that showed good coherence to the provided CRP potentials. The plot of the three dimensional potential surface also portrays a Morse function which is in correspondence to the model that was used to build the LEPS potentials. Even though, there are some abnormalities in the mean relative errors, we have obtained tolerable absolute errors that verifies the reliability of the program developed. This project could have been further developed to represent the modified Morse potential by obtaining the eight Morse potential parameters. However, due to time constraints and some lack of clarity in the first few parts of the project, this was not achieved.