

Molecular dynamics (MD) Simulations of Chemical Vapor
Deposition (CVD) of Carbon Dimer on a Diamond (100)
Surface and Application of Neural Networks (NN)
for Event Probability Predictions

By

ABDUL NIZAM ABDUL SAMADH

Bachelor of Science

University of Madras

Chennai - 600005

Tamilnadu, India

2001

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTER OF SCIENCE
July, 2005

Molecular Dynamics (MD) Simulations of Chemical Vapor
Deposition (CVD) of Carbon Dimer on a Diamond (100)
Surface and Application of Neural Networks (NN)
for Event Probability Predictions

Thesis Approved:

Dr. R. Komanduri

Thesis Advisor

Dr. L. M. Raff

Dr. M.T. Hagan

Dr. H. B. Lu

Dr. A. Gordon Emslie

Dean of the Graduate College

SUMMARY

Carbon dimers are found to be an important growth species in the growth of nanocrystalline diamond (NCD) through CVD process. Events, such as chemisorption, reflection, and desorption occur during the deposition of carbon dimers on to the substrate on which the diamond films are to be grown. The probabilities of each of these events have a significant effect on diamond growth. Molecular Dynamics (MD) simulations are widely used to predict the probabilities of such events. Though, MD simulations give agreeable results with experimental values, the calculation of the effect of different input parameters on various events involve time consuming numerical methods and hence the process is cumbersome. In this study, initially MD simulations of carbon dimer deposition on diamond (100) surface were performed using a many body empirical potential and the probabilities of the aforesaid events were calculated by varying the input conditions. This information was used to implement Neural Networks (NN) to predict the probabilities of the events. The neural network was also used to predict the underlying relationship between various input parameters and event probabilities. The computational time for the prediction of the events using molecular dynamics is generally several days while implementation of neural networks reduces it to mere minutes. The functional relationship between various input parameters and event probabilities predicted by NN is found to agree well with the MD simulation results.

ACKNOWLEDGMENTS

Though myriad events happen in one's life span, only a few remain precious and evergreen. Writing a thesis is one such event. I take immense pleasure in expressing my sincere thanks to all those who made this work a successful one.

First, I would like to convey my hearty thanks to my parents and sisters for all their love, support, encouragement and prayers that made me to reach new pinnacles in every walk of my life.

I would like to express my sincere thanks to the National Science Foundation (NSF) for lending their support to this project.

I wish to express my deepest gratitude to my advisor, Dr. Ranga Komanduri, for believing in my abilities and offering me a chance to work on this project. His passion for research and exploring new areas will forever be remembered. Unfortunately, I have no amount of words to convey my appreciation to him for his guidance, patience and support throughout my graduate career.

I am beholden to Dr. Lionel M. Raff, Dr. Paras M. Agrawal and Dr. Martin T. Hagan, for all their efforts and precious suggestions that made me to overcome many barriers during this project. Their ingenuity and ability to explain even the most intricate things in a simple manner have always fascinated me.

I am also grateful to Dr. Hong Bing Lu for being kind enough to serve on my thesis committee. I would like to thank Dr.Satish Bukkapatnam for his suggestions and ideas during the MD group meetings.

To my friend, Bala subramanian, thank you for all the pains and efforts you have taken in making me achieve my goal. Your ability in problem solving and passion for learning new things have always mesmerized me. I would also like to thank my friends Ramya, Siva and Freeman for their encouragement, time and help. Thanks to Rutu, Milind, Krishnaveni and other members of our MD group, for their useful suggestions and discussions.

Finally I would like to thank all my colleagues and friends who made my graduate career program a pleasant and memorable one.

TABLE OF CONTENTS

Chapter	Page
1. INTRODUCTION	1
1.1. Chemical Vapor Deposition	1
1.2. Molecular Dynamics Simulations	3
1.3. Neural Networks	4
2. ESSENTIALS OF MOLECULAR DYNAMICS SIMULATIONS	8
2.1. Introduction	8
2.2. Procedure for conducting MD Simulation	8
2.3. Interatomic Potentials	9
2.4. Time Integration Algorithm	16
2.5. Advantages and Limitations of MD Simulations	18
3. ESSENTIAL OF NEURAL NETWORKS	21
3.1. Introduction	21
3.2. Evolution of neural networks: Milestones and Developments	22
3.3. Neural network components and parameters	23
3.3.1. Neuron	23
3.3.2. Weights and Biases	24
3.3.3. Transfer functions	25
3.4. Neural Network Classification	27

3.4.1.	Single layer neural network	27
3.4.2.	Multilayer network	27
3.5.	Neural network learning and testing	28
3.6.	Feed forward neural networks	29
3.6.1.	Architecture of the feed forward neural network	29
3.6.2.	Working of the feed forward network	30
3.7.	Learning Algorithms	31
3.7.1.	Least-Mean-Squared (LMS) rule	31
3.7.2.	Back propagation algorithm	33
3.7.3.	Levenberg-Marquardt algorithm	37
4.	LITERATURE REVIEW	39
5.	PROBLEM STATEMENT	54
6.	MOLECULAR DYNAMICS SIMULATION OF CARBON DIMER (C₂) DEPOSITION ON DIAMOND (100) SURFACE	56
6.1.	Computational Model	56
6.2.	Parameters of Interest	59
6.2.1.	Incident polar angle	59
6.2.2.	Rotation angle	61
6.2.3.	Impact parameter	61
6.2.4.	Translational energy of the carbon dimer	62
6.2.5.	Rotational energy of the carbon dimer	63
6.3.	Predominant events in CVD dimer deposition	64
6.3.1.	Chemisorption	65
6.3.2.	Scattering	67

6.3.3.	Desorption	68
7.	NEURAL NETWORKS (NN) FOR EVENT PROBABILITY PREDICTION	70
7.1.	Architecture and working of the neural network	70
7.2.	Implementation of the neural network	71
8.	PREDICTION OF EVENT PROBABILITIES NEURAL NETWORK	
	Vs MD SIMULATIONS	73
8.1.	Data points generation for neural networks	73
8.2.	Training and testing of the event probability	
	Neural network	74
8.3.	Effect of input parameters on event probabilities:	
	Neural Network Vs MD predictions	78
	8.3.1. Effect of Incidence angle	78
	8.3.2. Effect of Rotation angle	80
	8.3.3. Effect of Impact Parameter	82
	8.3.4. Effect of Translational energy of the dimer	83
	8.3.5. Effect of Rotational energy of the dimer	85
8.4.	Statistical uncertainty : Neural network Vs MD	86
8.5.	More results from neural networks	88
9.	CONCLUSIONS AND FUTURE INVESTIGATIONS	94
9.1.	Conclusions	94
9.2.	Future Work	95
	REFERENCES	

LIST OF TABLES

Table	Page
2.1. Parameters for carbon-carbon pair terms	12
2.2. Parameters for the angular contribution to the carbon bond order	13
2.3. Parameters needed for carbon-carbon cubic spline	15
2.4. LJ potential parameter values	16

LIST OF FIGURES

Figure	Page
3.1. Single input neuron	24
3.2. Tan sigmoid transfer function	26
3.3. Purelin transfer function	26
3.4. Single layer neural network	27
3.5. Multilayer neural network	28
3.6. Feed forward neural network	29
4.1. Six types of chemisorption configurations on diamond (001) - (2 x 1) surface.	45
6.1. Simulation model and carbon dimer (C ₂)	58
6.2. Top three layers of atoms of diamond (100) substrate and radical site	58
6.3. Sketch showing input variables for the trajectory calculations	60
6.4. Distribution of incidence angle	60
6.5. Distribution of rotation angle	61
6.6. Distribution of impact parameter	62
6.7. Distribution of translational energy of the dimer	63
6.8. Distribution of rotational energy of the dimer	64
6.9. Variation of systems potential energy, Z coordinate of COM of the	

dimer, distance between carbon atoms of the dimer (chemisorption event)	66
6.10. Variation of systems potential energy and Z coordinate of COM of the dimer (scattering event)	67
6.11. Variation of systems potential energy and Z coordinate of COM of the dimer (desorption event)	69
8.1. Neural network training and testing plots for the probability of chemisorption for one neural network	75
8.2. Neural network training and testing plots for the probability of scattering for one neural network	76
8.3. Neural network training and testing plots for the probability of desorption for one neural network	77
8.4. Effect of incidence angle on chemisorption, scattering and desorption probabilities – MD and Neural network predictions. The error bars represent one sigma limit of statistical uncertainty in the MD results.	79
8.5. Effect of rotation angle on chemisorption, scattering and desorption probabilities – MD and Neural network predictions. The error bars represent one sigma limit of statistical uncertainty in the MD results.	81
8.6. Effect of impact parameter on chemisorption, scattering, and desorption probabilities – MD and Neural network predictions. The error bars represent one sigma limit of statistical uncertainty	

in the MD results.	82
8.7. Effect of translational energy on chemisorption, scattering and desorption probabilities – MD and Neural network predictions. The error bars represent one sigma limit of statistical uncertainty in the MD results.	84
8.8. Effect of rotational energy on chemisorption, scattering and desorption probabilities – MD and Neural network predictions. The error bars represent one sigma limit of statistical uncertainty in the MD results.	85
8.9. Comparison of statistical uncertainty in neural network and MD	87
8.10. Effect of impact parameter on event probabilities for various translational energies of the dimer- Neural network predictions	89
8.11. Effect of translational energy on event probabilities for various Incidence angle of the dimer- Neural network predictions	90
8.12. Effect of rotation angle on event probabilities for various translational energies of the dimer- Neural network predictions	91
8.13. Effect of incidence angle on event probabilities for various Impact parameters - Neural network predictions	92
8.14. Effect of rotational energy on event probabilities for various Impact parameters - Neural network predictions	93

CHAPTER 1

INTRODUCTION

1.1. Chemical Vapor Deposition

Diamond is the hardest material known. Its unique mechanical, chemical, and electrical properties make it not only one of the most scientifically and technologically valuable material but also one of the most fascinating material known to researchers. It was in the mid-1950s that diamond was first successfully synthesized on a commercial scale using high pressure, high temperature (HP-HT) techniques [1]. Efforts were made about the same time to grow diamond directly from gases, but since the growth rates were extremely low, the vapor-phase deposition of diamond was not assigned much importance. In the early 1980's, when it was shown that growth rates in the range of a few $\mu\text{m}/\text{hour}$ can be obtained using vapor phase deposition [2-5], this technique became an area of general attraction and exploration among researchers.

Chemical Vapor Deposition (CVD) is a vapor phase deposition technique in which the gaseous reactants undergo chemical reaction in an activated environment, such as plasma, leading to the formation of stable solid product, such as diamond powders or thin diamond films on the surface of the heated

substrate. In a CVD diamond film growth, some of the widely used gaseous species include CH_3 , C_2H_2 , C_2H_4 , and C_2 . The CVD process has a wide range of advantages over other thin-film deposition processes. A few such advantages are the ability to produce highly dense and uniform films with good reproducibility and adhesion, reasonable deposition rates and processing cost, ability to control surface morphology and orientation of the films obtained by controlling the CVD parameters, ability to adjust the deposition rates easily and flexibility of using a wide range of chemical precursors, such as halides, hydrides which enable the deposition of wide variety of films apart from the diamond films [6]. The diamond produced by the CVD process is comparable in purity and properties to HP-HT or natural diamond that makes it a potential candidate for numerous applications. The CVD produced diamond films are used as coatings on cutting tool inserts to enhance the tool life, protective windows or optical coatings with high transmittance in the visible and infrared region, as shadow mask supports in x-ray lithography of electronic components [7], etc. The process of diamond film growth by CVD process involves a number of complex reaction mechanisms taking place between the surface atoms of the substrate and the gaseous species. There are a few elementary reactions, such as chemisorption, insertion, scattering, and desorption that serve as the building blocks for the complex reactions leading to thin-diamond film growth. Hence, the investigations of these preliminary reaction events, their probabilities, various parameters

affecting the probabilities of their occurrence have gained significant importance [8-11]. Molecular dynamics simulations is one of the most powerful and widely used tools for investigating such reaction mechanisms and reaction events occurring during diamond film growth in a CVD process.

1.2. Molecular Dynamics Simulations

Molecular dynamics (MD) simulation, as the term indicates, deals with simulating the behavior of a system at the atomistic level under given processing conditions. In MD simulations, the system is represented as an ensemble of atoms. In such simulations, we make use of a potential energy function which gives the potential energy experienced by each atom due to its position relative to that of its neighbors. From this potential energy, we can determine the force experienced by each atom as time progresses. Molecular dynamics simulation is a deterministic approach, where, once the current positions of the atoms as well as the forces acting on the atoms due to their neighbors are known then the positions of the atoms after a very small time increment, (usually in the orders of femtosecond) can be easily evolved by integrating the Newtonian equations of motion using suitable time integration algorithms. In spite of its high computational cost, today MD simulations serve as a powerful tool in the study of nanometric cutting [70-71], different types of fracture mechanism [72], film-growth mechanisms [73-74], friction and surface property studies [75], and biomaterial engineering [76].

Carbon dimers are found to be an important reaction species in the growth of nanocrystalline diamond films using CVD process [27-30]. Therefore the

investigation of various reaction channels and mechanism that occur when carbon dimer is used as growth species is always of immense interest among researchers [8]. In this study, MD simulations have been used to study the probabilities of various elementary gas phase reactions, such as chemisorption, desorption, insertion and scattering occurring during diamond-film growth in a CVD process when carbon dimmers (C_2) are used as the growth species. The results from the MD simulations are used to train a neural network (NN) and then that neural network is used to predict the probabilities of various events.

1.3. Neural Networks

A neural network (NN) is an artificial network of neurons that mimics or emulates the real network of neurons present in the human brain. Though the neural networks are not as sophisticated as the networks present in the human brain, they have the capability to predict many complex underlying functions between variables of any particular process or event.

A neural network basically consists of a number of artificial neurons or nodes, typically arranged in layers, interconnected through a set of links. Each link multiplies its input by a suitable parameter called the weight before supplying it to the next neuron. Each neuron sums over its input and passes the output, which is a weighted sum of the input and the bias, to a suitable transfer function. The output from this transfer function is the final output. This output can be made the inputs to the next layer of neurons. The network used in this study is a multilayered feed-forward network, which has two layers, of which the first layer

is called a hidden layer because its input and output are not available to the outside world. The hidden layer is associated with a tan sigmoid transfer function that gives the network the ability to learn the linear and nonlinear relations that exist between the input and outputs. It also makes them an ideal choice for generalization and event probability predictions.

We have two stages in the implementation of a neural network. The first stage is the learning stage and the second stage is the testing or the production stage. During the learning stage, the neural network is provided with a set of input data as well as the corresponding output data and the network is made to learn by examples. The difference between the neural network output and the actual target output is used to determine the error which is used in strengthening the network so that its subsequent predictions are better. Once the error from neural network prediction has been sufficiently reduced, the network is assumed to have completed its learning phase. The network is then subjected to the next stage, namely, the testing stage. In this, the neural network is provided with a set of input data. Corresponding output data are not given to the network. The network is allowed to make its choice or prediction based on its previous experience in encountering such data during the training. If the network is able to predict the corresponding output data correctly, then it indicates that the network has been properly trained and has attained the ability.

In this study we have used neural networks to predict the probabilities of various events occurring in a CVD process when a carbon dimer (C_2) is used as the species for diamond film growth. The input parameters, namely, incidence

angle (θ), rotation angle (Φ), impact parameter (b), translational energy (E_{Trans}), and rotational energy (E_{Rot}) form the input to the neural network. The probabilities of events, such as chemisorption, scattering, and desorption form the output of the neural network. The network is first trained by supplying the input parameters and the corresponding event probabilities. After the neural network has been trained well, the network is made to predict the event probabilities for a given set of input parameters.

The conventional approach adopted so far by chemists is to use MD simulations for predicting the event probabilities for a given set of input conditions [32-33]. But, the drawback of MD simulations is that it involves computationally intensive and time consuming numerical integration algorithms. So, the exploration of the entire range of input parameters becomes a very difficult task, and also the time taken increases with the number of atoms comprising the system. It will be shown in this investigation that using neural networks the computational time for determining the effect of various input parameters on event probabilities can be reduced from hours to mere minutes. Also, this procedure is independent of the number of atoms comprising the system, thereby giving us an opportunity to explore a wide range of input data values.

In this investigation, we initially ran MD trajectories for different sets of input conditions and determined the event probabilities which were then used for training the neural network. Chapter 2 will cover the empirical potentials used, the time integration algorithms, and the advantages and limitations of MD

simulations. Chapter 3 deals with the basic components of neural networks, their classifications, and stages in the implementation of neural network, different training algorithms used. Chapter 4 reviews the literature on various kinds of species employed in a CVD diamond film growth, MD simulations performed on various surfaces, such as diamond and silicon surfaces, time involved in such studies, and how neural networks have been effectively used in a CVD thin film growth processes. Chapter 5 discusses the drawbacks of using MD simulation for event probability predictions and the solution to overcome the present situation. Chapter 6 deals with the distributions for the five input parameters, and the event probabilities considered in this investigation. Chapter 7 presents the architecture of the network employed in this investigation and the implementation of the neural network. Chapter 8 deals with the neural network predictions of various event probabilities for different sets of input conditions and the comparison of neural network prediction with MD simulation results. Also this chapter discusses the statistical error involved in MD simulations as against the error in neural network predictions. Chapter 9 presents the conclusions based on the results of this investigation and proposes future investigations that can be carried over using the current neural network technique that has been implemented in this investigation.

CHAPTER 2

ESSENTIALS OF MOLECULAR DYNAMICS (MD) SIMULATIONS

2.1. Introduction

In molecular dynamics (MD) simulations, the entire workpiece is represented as an ensemble of atoms. Given the initial positions and forces acting on the atoms, the subsequent positions and forces on the atoms can be evolved over time by integrating Newton's equations of motion using a suitable time integration algorithm [12]. This chapter mainly focuses on the basic procedures for carrying out the molecular dynamics simulation, the advantages and disadvantages of MD, the interaction potential, and the time integration algorithm used in this study.

2.2. General Procedure for conducting MD simulations

In molecular dynamics simulations, all the atoms in the system are considered as point masses. The initial position of the atoms are selected based on the structure of the system under consideration. The detailed procedure for carrying out MD simulation is as follows:

1. Based on the initial position of each atom with respect to its neighboring atoms, the potential energy (V) experienced by the system at an initial time, say t_0 , is determined using an empirical potential.

2. The forces (F) acting on an atom are then determined by taking the derivative of the potential (V) with respect to the position (r) of the atom,

$$F = -\nabla V = -\frac{dV}{dr} \quad . \quad (2.1)$$

3. Newton's second law can be mathematically expressed as follows

$$F = ma \quad , \quad (2.2)$$

where ' m ' is the mass of the atom and ' a ' is its acceleration. The acceleration of the atom can be determined from Eqns. (2.1) and (2.2).

4. Once the acceleration of the atom at time ' t_0 ' is known, the new velocity (v_{new}) and the new position (r_{new}) of the atom, after an infinitesimal time period of δt , can be calculated using the following equations

$$v_{new} = a\delta t + v_{initial} \quad , \quad (2.3)$$

$$r_{new} = r_{initial} + v_{new}\delta t \quad , \quad (2.4)$$

where $v_{initial}$ represents the initial velocity and $r_{initial}$ the initial position of the atom. The infinitesimal time period used in MD simulations is usually on the order of a few femto seconds.

5. Finally, the atoms are displaced to their corresponding new positions calculated above and again the steps stating from 1 through 5 are repeated to monitor the evolution of the system with time under the given operating conditions.

2.3. Interatomic Potentials

Interaction potentials form the main ingredient of MD simulations. A potential is a function of relative positions of atoms with respect to each other,

representing the potential energy of the system for a given configuration of the atoms comprising the system. The interatomic potential functions are both rotationally and translationally invariant. These functions are usually derived empirically and hence, are known as empirical potentials. A number of potentials exist today, some to mention are the Morse potential, Stillinger-Weber potential, Tersoff potential, Lennard Jones potential, and Brenner Potential [13-14].

The potential used in this study for short range interaction is a many-body potential given by Brenner *et al.* [13].

The potential, V can be written as a sum over atomic sites i ,

$$V = \frac{1}{2} \sum_i E_i, \quad (2.5)$$

where each contribution of E_i is given by

$$E_i = \sum_{j(\neq i)} [V_R(r_{ij}) - B_{ij} V_A(r_{ij})] \quad . \quad (2.6)$$

In Eqn. (2.6), the summation is over the nearest neighbors j of atom i , excluding atom i , B_{ij} is the many body coupling term between the bond from atom i to atom j and the local environment of atom i , $V_R(r)$ and $V_A(r)$ represents the pair-additive repulsive and attractive interactions. Eqn. (2.5) given by Abell-Tersoff [14] can realistically describe carbon-carbon single, double, and triple bond lengths and energies in hydrocarbons and in solid graphite and diamond. However, the problem with this expression is that the assumption of near-neighbor interactions combined with the sum over atomic sites results in nonphysical behavior in the case of intermediate bonding situations. Nonphysical behavior arises again when conjugated and nonconjugated bonds are examined.

The aforesaid problems have been overcome by rewriting Eqns. (2.5) and (2.6) as follows

$$V = \sum_i \sum_{j(>i)} [V^R(r_{ij}) - \bar{b}_{ij} V^A(r_{ij})] \quad . \quad (2.7)$$

Eqn. (2.7) represents the potential given by Brenner *et al.* [13] and is a modified form of Abell-Tersoff [14] potential function. Here, V represents the interaction potential, $V^R(r_{ij})$ and $V^A(r_{ij})$ are pair-additive interactions that represent all interatomic repulsions (core-core) and attraction, r_{ij} is the distance between pairs of nearest-neighbor atoms i and j , and \bar{b}_{ij} is the bond order between atoms i and j and is conveniently represented as follows

$$\bar{b}_{ij} = \frac{1}{2} [b_{ij}^{\sigma-\pi} + b_{ji}^{\sigma-\pi}] + b_{ij}^{\pi} \quad . \quad (2.8)$$

Values of $b_{ij}^{\sigma-\pi}$ and $b_{ji}^{\sigma-\pi}$ depend on the local coordination and bond angles for atom i and j . The term b_{ij}^{π} can be expressed as

$$b_{ij}^{\pi} = \pi_{ij}^{RC} + b_{ij}^{DH} \quad . \quad (2.9)$$

The terms indicating the interatomic repulsions and attractions are given by

$$V^R(r) = f^c(r)(1 + Q/r)Ae^{-\alpha r} \quad , \quad (2.10)$$

$$V^A(r) = f^c(r) \sum_{n=1,3} B_n e^{-\beta_n r} \quad . \quad (2.11)$$

Table 2.1. Parameters for carbon-carbon pair terms used in Eqns. (2.10) and (2.22).

$B_1 = 12388.79197798 \text{ eV}$	$\beta_1 = 4.7204523127 \text{ \AA}^{-1}$	$Q = 0.3134602960833 \text{ \AA}$
$B_2 = 17.56740646509 \text{ eV}$	$\beta_2 = 1.4332132499 \text{ \AA}^{-1}$	$A = 10953.54416217 \text{ eV}$
$B_3 = 30.71493208065 \text{ eV}$	$\beta_3 = 1.3826912506 \text{ \AA}^{-1}$	$A = 10953.544162170 \text{ eV}$
$D_{\min} = 1.7$	$D_{\max} = 2.0$	

The parameters used for the carbon-carbon pair terms in Eqns. (2.10), (2.11), and (2.22) are given in Table 2.1.

The first term in Eqn. (2.8) is given by

$$b_{ij}^{\sigma-\pi} = [1 + \sum_{k (\neq i, j)} f_{ik}^c(r_{ik}) G(\cos(\theta_{ijk})) e^{\lambda_{ijk}} + P_{ij}(N_i^C, N_i^H)]^{-1/2}. \quad (2.12)$$

The subscripts i and j refer to the atom identity, the function P represents a bicubic spline. The function $f^C(r)$ ensures that the only nearest neighbors are included in the interactions. It limits the range of covalent interactions. N_i^C and N_i^H represent the number of carbon and hydrogen atom neighbors of atom i and are represented as:

$$N_i^C = \sum_{k (\neq i, j)}^{\text{carbon atoms}} f_{ik}^c(r_{ik}), \quad (2.13)$$

$$N_i^H = \sum_{l (\neq i, j)}^{\text{Hydrogen atoms}} f_{il}^c(r_{il}). \quad (2.14)$$

The values of λ and function P are taken to be zero for solid state carbon. An expression for $b_{ji}^{\sigma-\pi}$ can be obtained by interchanging the subscripts in Eqn. (2.12). The function $G(\cos(\theta_{ijk}))$ in Eqn. (2.12) controls the contribution each

nearest neighbor makes to the empirical bond order according to the cosine of the angle of the bonds between atoms i and k and atoms i and j . The parameters for the angular contribution to the carbon bond order are given in Table 2.2.

Table 2.2. Parameters for the angular contribution to the carbon bond order

θ (rad)	$G(\cos(\theta))$	$dG/d(\cos(\theta))$	$d^2G/d(\cos(\theta))^2$	$\gamma(\theta)$
0	8	-	-	1
$\pi/3$	2.0014	-	-	0.416335
$\pi/2$	0.37545	-	-	0.271856
0.6082π	0.09733	0.4	1.98	-
$2\pi/3$	0.05280	0.17	0.37	-
π	-0.001	0.104	0.00	-

The term π_{ij}^{RC} in Eqn. (2.9) represents the influence of radical energetics and π -bond conjugation on the bond energies. This term takes care of correctly describing the radical structures in diamond and accounts for non-local conjugation effects in graphite and benzene. This term was absent in the first generation form of the Brenner Potential [15]. The term π_{ij}^{RC} is taken as a tricubic spline F

$$\pi_{ij}^{RC} = F_{ij}(N_i^t, N_j^t, N_{ij}^{conj}) \quad , \quad (2.15)$$

that depends on the total number of neighbors of bonded atoms i and j , as well as a function N_{ij}^{conj} that depends on local conjugation.

The term N_i^t represents the coordination of atom i given by

$$N_i^t = N_i^C + N_i^H \quad . \quad (2.16)$$

The function N_{ij}^{conj} is represented as

$$N_{ij}^{conj} = 1 + \left[\sum_{k(\neq i, j)}^{\text{carbon}} f_{ik}^c(r_{ik}) F(X_{ik}) \right]^2 + \left[\sum_{l(\neq i, j)}^{\text{carbon}} f_{jl}^c(r_{jl}) F(X_{jl}) \right]^2, \quad (2.17)$$

where

$$\begin{aligned} F(x_{ik}) &= 1, & x_{ik} &< 2 \\ F(x_{ik}) &= [1 + \cos(2\pi(x_{ik} - 2))]/2, & 2 &< x_{ik} < 3 \\ F(x_{ik}) &= 0, & 3 &< x_{ik} \end{aligned} \quad (2.18)$$

and

$$x_{ik} = N_k^t - f_{ik}^c(r_{ik}) \quad . \quad (2.19)$$

The value of N_{ij}^{conj} becomes 1 if all the neighbors bonded to a pair of carbon atoms i and j have four or more neighbors and the bond between these atoms is considered to be part of a conjugated system. N_{ij}^{conj} becomes greater than 1, if the coordination number of the neighboring atoms decrease, indicating a conjugated bonding configuration.

The term b_{ij}^{DH} in Eqn. 2.9 is given by

$$b_{ij}^{DH} = T_{ij}(N_i^t, N_j^t, N_{ij}^{conj}) \left[\sum_{k(\neq i, j)} \sum_{l(\neq i, j)} (1 - \cos^2(\Theta_{ijkl})) f_{ik}^c(r_{ik}) f_{jl}^c(r_{jl}) \right], \quad (2.20)$$

where

$$\Theta_{ijkl} = e_{jik} e_{ijl} \quad . \quad (2.21)$$

The function $T_{ij}(N_i^t, N_j^t, N_{ij}^{conj})$ is a tricubic spline, and the functions e_{jik} and e_{ijl} are unit vectors in the direction of the cross products $R_{ji} \times R_{ik}$ and $R_{ij} \times R_{jl}$, respectively, where the R 's are vectors connecting the subscripted atoms. Table 2.3. gives the values needed for the carbon-carbon cubic spline T in Eqn. 2.20. All function values and derivatives not given in the table are equal to zero.

Table 2.3. Parameters needed for carbon-carbon cubic spline T in Eqn. (2.20).

i	j	k	T(i, j, k)	Fitting data/Structure
2	2	1	-0.070280085	Ethene
2	2	9	-0.00809675	Solid-state structure

The entire parameter-fitting method discussed above was made much easier by assuming only nearest-neighbor interactions. However, the best way to define this for a continuous function is problematic. The value of $f_{ij}^c(r)$ is defined by a switching function of the form

$$\begin{aligned}
 f_{ij}^c(r) &= 1, & r &< D_{ij}^{\min} \\
 f_{ij}^c(r) &= \frac{1}{2} \left[1 + \cos((r - D_{ij}^{\min}) / (D_{ij}^{\max} - D_{ij}^{\min})) \right], & D_{ij}^{\min} &< r < D_{ij}^{\max} \\
 f_{ij}^c(r) &= 0, & r &> D_{ij}^{\max}
 \end{aligned} \tag{2.22}$$

where $D_{ij}^{\max} - D_{ij}^{\min}$ defines the distance over which the function goes from one to zero.

The Van Der Waal (VDW) interaction between the atoms comprising the system is taken care of by the L-J (12-6) potential. The L-J (12-6) potential has the following form

$$V(r) = 4\epsilon \left[\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right].$$

The well depth ϵ and the equilibrium separation r are the only adjustable parameters in the L-J (12-6) potential. In this investigation, the values of the well depth and equilibrium separation for the VDW interaction between two atoms are given in Table 2.4

Table 2.4. L-J potential parameter values

Atoms Type	Well depth (meV)	Equilibrium separation (Å)
Carbon - Carbon	4.412	2.28
Carbon - Hydrogen	1.806	2.54
Hydrogen - Hydrogen	0.740	2.81

2.4. Time Integration Algorithm

In MD simulations we integrate the equations of motion over a given time period using numerical integration techniques. These time integration algorithms are based on finite difference methods. Here, the total time is discretized into a finite number of equal time intervals or time steps given by Δt , which is 0.5 fs in the present investigation. If the positions and their time derivatives at time t are known, the integration algorithm gives the same quantities at a later time, $t + \Delta t$. The integration algorithms can show the evolution of the system with time by

repeating the procedure mentioned above. Currently, there exist a number of integration algorithms, such as the Verlet algorithm [16], Beeman algorithm [17]. The algorithm used in the present investigation for integrating the equations of motion is the “Gear Predictor-Corrector” algorithm [18]. The algorithm consist of two parts, namely, the predictor part and the corrector part.

Predictor Part:

If we know the position r , velocity v , acceleration a , and some other time derivatives up to a certain degree q at a given time t , the Taylor expansion can be used to predict the values of these quantities at time $t + \Delta t$. The newly predicted values of the position, velocity, acceleration and the rate of change of acceleration after a time interval Δt is given by:

$$r^p(t + \delta t) = r(t) + \delta t v(t) + (1/2) (\delta t)^2 a(t) + (1/6)(\delta t)^3 b(t) + \dots , \quad (2.23)$$

$$v^p(t + \delta t) = v(t) + \delta t a(t) + (1/2) (\delta t)^2 b(t) + \dots , \quad (2.24)$$

$$a^p(t + \delta t) = a(t) + \delta t b(t) + \dots , \quad (2.25)$$

$$b^p(t + \delta t) = b(t) + \dots , \quad (2.26)$$

where r^p , v^p , a^p , and b^p represents the position, velocity, acceleration, and rate of change of acceleration after a time interval of Δt from the initial time interval t .

Force Calculation:

The force on the atom is calculated by taking the derivative of the potential with respect to the position of the atom and is given by

$$F = -\nabla V_i^p = -\frac{dV(r^p)}{dr^p} . \quad (2.27)$$

From the force, we can calculate the acceleration using Eqn. (2.2). The resulting value of acceleration will be different from that predicted by the above Taylor's expression. The difference between these two values constitutes an "error signal" given by

$$\Delta a(t+\delta t) = a^c(t+\delta t) - a^p(t+\delta t) \quad . \quad (2.28)$$

Corrector Part:

The "error signal" along with certain other coefficients is used to correct the values of the position, velocity, and acceleration predicted by the predictor method. The corrected values of the position, velocity, acceleration, and the rate of change of acceleration at time $t + \Delta t$ are given by

$$r^c(t+\delta t) = r^p(t+\delta t) + c_0 \Delta a(t+\delta t) \quad , \quad (2.29)$$

$$v^c(t+\delta t) = v^p(t+\delta t) + c_1 \Delta a(t+\delta t) \quad , \quad (2.30)$$

$$a^c(t+\delta t) = a^p(t+\delta t) + c_2 \Delta a(t+\delta t) \quad , \quad (2.31)$$

$$b^c(t+\delta t) = b^p(t+\delta t) + c_3 \Delta a(t+\delta t) \quad , \quad (2.32)$$

where c_0, c_1, c_2 , and c_3 are the coefficients of proportionality given by [16]:

$c_0 = 1/6, c_1 = 5/6, c_2 = 1$, and $c_3 = 1/3$.

2.5. Advantages and Limitations of MD simulations

Molecular dynamics simulation is one of the powerful tool widely used in the study of very complex reaction mechanisms. Some of the strengths of MD simulation that has made it to take a leading-edge method over other techniques are the following [12]:

1. Molecular dynamics simulations offer a great opportunity to explore the behavior of systems at atomistic and molecular levels.

2. MD simulations can be of immense use in simulating experiments that are very costly and difficult to carry in real world. For example, when MD is applied to experiments, such as nanometric machining, the effect of various parameters such as tool geometry and cutting speed can be easily studied at an insignificant fraction of the cost.
3. When simulations are running, the human involvement required is almost not there unlike the real world experiments where utmost care has to be taken when the experiments are in progress.
4. A major advantage of MD simulation is its repeatability. Any particular simulation can be exactly repeated any number of times with the same degree of accuracy.
5. MD simulation is a very deterministic technique, providing complete information, such as potential energy, velocity and force experienced by each and every atom comprising the system at any point of time which can be easily and accurately evolved.

Though MD simulation has numerous advantages as stated above, it also has its own limitations, a few of these limitations are given in the following:

1. In MD simulations, results are purely dependent on the forces acting between atoms based on their positions. These forces are obtained by taking the derivative of the empirical potential function. Therefore, the extent to which molecular dynamics simulation can imitate real experiments depends on the ability of the potential function to reproduce the real behavior of the system.

2. MD simulations involve integration of Newton's equations to obtain new velocities and positions of the atoms. The integration involves the use of numerical algorithms that require a very small integration timestep to give accurate results. Therefore, MD simulations are not preferred to simulate processes or reactions that take very long time periods.
3. The computational time and costs involved in MD simulations are significant because Newton's equations of motion are to be integrated for every atom comprising the system and for each time step. The time and costs increase rapidly with increase in the size of the system considered.
4. When the temperature of the system considered is very low, quantum effects become significant. In such cases MD simulation results have to be interpreted with utmost caution due to the possibility of errors in the potential used.

CHAPTER 3

ESSENTIALS OF NEURAL NETWORKS (NN)

3.1. Introduction

A neural network (NN) can be considered as a computational system made up of a number of simple and highly interconnected processing elements called nodes, which processes information by its active state of response to external inputs [69]. The structure and working of the human brain serves as the basic inspiration for the invention and development of neural networks[69]. Neural network attempts to emulate the adaptability, intelligent decision making and information processing ability of the brain. The greatest strength of neural networks is adaptive learning. It has the capability to learn, generalize, and reproduce from experience and examples.

First, the neural network is trained using a number of examples and then the network is tested to see whether it can interpret new data based on previous experience. Neural networks offer a wide range of advantages, such as adaptive learning, self-organization, fault tolerance, and easy implementation that allow them to take a lead over other approaches.

The neural network architecture, terminologies, training algorithms, and methodologies followed in this investigation were adopted from the book Neural Network Design Hagan *et al.* [69]. The book presents the most useful and

practical NN architectures, learning rules and training algorithms in a clear and consistent manner. Various topics of practical importance in the application of neural networks and neural network operations have been well explained in this book [69].

3.2. Evolution of neural networks: milestones and development

Neural networks is a field of recent origin. However, this field has a long history tracing back to periods before the invention of computers and has survived at least one major setback and several decades of oblivion. McCulloch and Pitts [59] used formal logic to create neural network models with simple neurons that were considered as binary devices with fixed thresholds. Their network was used mainly for simple logic functions such as “OR” and “AND”. Farley and Clark [60] created the first computer simulations of neuronal models and used normalization procedures to ensure better operation of their simulation models. Rosenblatt [61], a psychologist, designed and developed a three-layered system known as Perceptron network that exhibited adaptive behavior. Though Rosenblatt’s design was considered a milestone in the field of neural network, it had some limitations such as inefficiency in solving pattern recognition problems and inability to handle large inputs.

Widrow and Hoff [62] developed the ADALINE (ADaptive LINear Element) and MADALINE (Many ADALINEs) networks that employed a learning procedure called Least-Mean-Squared (LMS) learning rule. The network operates by attempting to minimize the difference between the observed and desired output. Amari [63] published a mathematical model that served as the basis for error-

correction methods employed in adaptive pattern classifications. Webros [64] first proposed the back-propagation algorithm that gave rise to back-propagation networks which are basically perceptrons with multiple layers with enhanced robustness and learning rules. Fukushima [65] developed competitive networks called Cognitron and Neocognitron for interpreting the handwritten characters.

Klopf [66] developed the “drive-reinforcement learning” for artificial neurons. This is similar to neuronal learning called “heterostasis” that occurs in biological neurons. Rumelhart and McClelland popularized the back-propagation algorithm in their book *Parallel Distributed Processing* [67]. Hopfield and Tank [68] developed the well known auto-associative network, the Hopfield Network, which attracted much attention due to its stability and ease of its fabrication using VLSI technology. Hagan *et al.* [69] introduced Gaussian-Newton approximation to Bayesian Regularization (GNBR) algorithm that reduced the cost of implementing the changes in the training algorithm and also produced optimal results with minimum computational time. Today, neural network concepts have been implemented on chips and are emerging as a prime solution to various complex problems representing the dominance of neural network in today’s scientific world.

3.3. Neural network components and parameters

3.3.1. Neuron

Biological neurons are the building blocks in the human brain. Likewise the artificial neuron forms the fundamental data processing unit of the neural network. Figure 3.1 shows a simple neuron model.

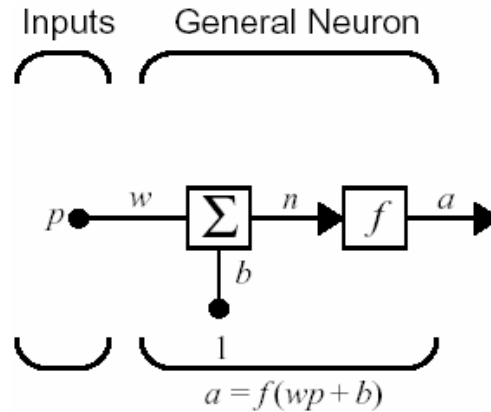


Fig. 3.1: Single input neuron [69]

Every neuron is associated with particular inputs, weights, biases, transfer functions, and outputs. The input ' p ' is first multiplied (weighted) by a suitable weight ' w ', and is passed on to the summer, the summer adds the weighted input ' wp ' with suitable bias ' b ' and passes the net input ' n ' to the transfer function ' f ', which operates on the net input ' n ' and produces an output ' a '. This output can be made to become the input to the next layer of neurons.

3.3.2. Weights and Biases

Every input supplied to a neuron is weighted before it is passed on to the summer in every neuron. Weights are a set of numbers associated with each interconnection between neurons in different layers. The weights indicate the strength of the interconnection between a neuron in one layer and another neuron in the next layer of the network. The initial values of the weights are set to zeroes or any small random number and are modified suitably during the network training to get the desired output from the network. Once the network has been completely trained, the final values of the weight matrix are stored and recurrently called during the testing session.

The bias can be considered as a threshold value added to the weighted inputs before they are passed on to the transfer function of the neuron. As can be seen from Figure 3.1, the bias has the effect of shifting the center of the transfer function f , while the weight changes the slope.

3.3.3. Transfer functions

Every artificial neuron in a neural network is characterized by its transfer function. Two neurons which are fed with the same inputs can produce different outputs depending on the transfer functions to which they are associated. A neuron can take many input signals, multiply by the weights, add the bias, and pass the resulting scalar on to the associated transfer function. The transfer function decides how the neuron will scale its response to the input data, and generates the neuron's activation.

Some of the transfer functions that are of frequent use in the neural network are the following:

- Hard-limit transfer function
- Linear transfer function
- Sigmoid transfer function

The neural network used in this study uses a tangent sigmoid (tansig) transfer function in the hidden layer, and a pure-linear transfer function in the output layer. The use of these transfer functions allow the network to understand the linear and non linear relationships that exist between it's input vectors and output vectors. The two transfer functions employed in this investigation are described in the following.

a. Hyperbolic tangent sigmoid transfer function (*tansig*)

The hyperbolic tangent sigmoid function, also known as the tan sigmoid transfer function takes input values between $-\infty$ and $+\infty$, produces an output signal between -1 and $+1$.

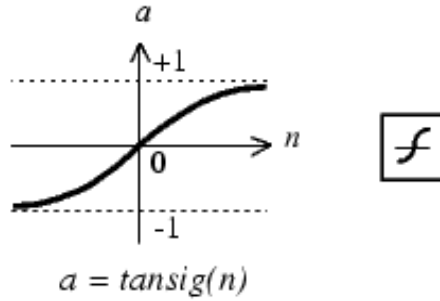


Fig 3.2: Tan sigmoid transfer function [69]

The output is calculated using the equation

$$a = \frac{2}{(1 + \exp(-2 * n))} - 1 \quad . \quad (3.1)$$

b. Pure-linear transfer function (*purelin*)

The pure-linear transfer function produces an output, linearly increasing with the input supplied to it. The pure-linear transfer function takes the following form

$$a = n \quad . \quad (3.2)$$

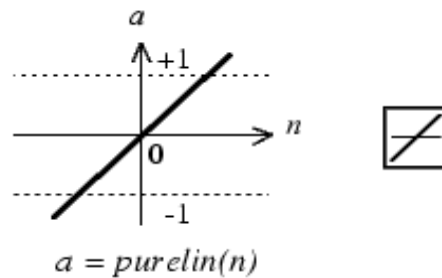


Fig 3.3: Purelin transfer function [69]

3.4. Neural network classification

Neural networks are classified into two major categories, namely, single-layered and multi-layered networks based on the number of layers in the network. Multilayer neural networks have sub classifications, such as multilayer feedforward networks and multilayer cooperative networks. The single-layer networks can also be subdivided into single-layer laterally-connected networks and single-layer topologically ordered networks.

3.4.1. Single-layer network

The single layer neural networks have only one layer of neurons. A single-layer neural network can have one or more neurons in their single layer and can produce one or more outputs. Figure 3.4 shows a single-layer neural network, having R inputs and S number of neurons

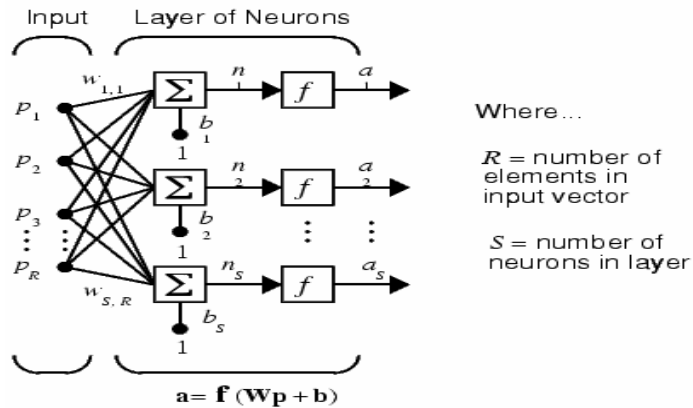


Fig 3.4: Single-layer neural network [69]

3.4.2. Multilayer network

Multilayer neural networks have more than one layer of neurons. In the multilayered feedforward network, which has been used in this study, all neural responses flow in a forward direction through different layers of the network.

Figure 3.5 below shows a multilayered feed forward network having a single

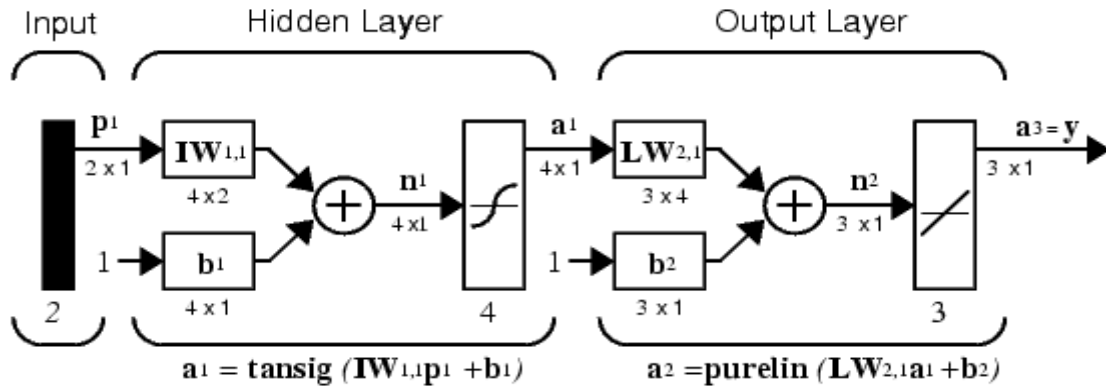


Fig 3.5: Multilayer neural network [69]

hidden layer with a sigmoid transfer function that gives the network the ability to learn linear and nonlinear relations that exist between inputs and outputs and an output layer with pure-linear transfer function allowing the neural network to produce values inside the range -1 to +1. The multilayered networks are an ideal choices for nonlinear regression and pattern recognition.

3.5. Neural network learning and testing

Basically there are two important stages in the implementation of neural networks for any application, namely, learning and testing. The feed forward network used in this study employs a supervised learning procedure. In supervised learning, the network is provided with a set of input vectors, 'A' as well as with corresponding desired output vectors, 'B'. During the learning process, the network compares its output vector, 'C', with the desired output vectors 'B' to produce the error percentage. The values of the weight matrix are adjusted so as to decrease the error. A network is said to be trained if its output responses are matching well with the desired outputs with minimum percent error.

After the completion of learning, the network is supplied with a testing data set which was never seen by the network during training and the output of the network is compared with the actual output to predict the network performance.

3.6. Feed forward neural networks

In the feed forward networks the flow of data always occurs in the forward direction from the input to the neurons in the hidden layers and thereon to the output layers. No information is back propagated during the operation of the network. Generally, the multilayered feed forward networks are associated with one or more hidden layers having sigmoidal functions that allow the network to learn both nonlinear and linear relationships between input and output vectors.

3.6.1. Architecture of the feed forward neural network

The multilayered feed forward network shown in Figure 3.6 has a total of three layers of neurons of which the first and second layers are known as the

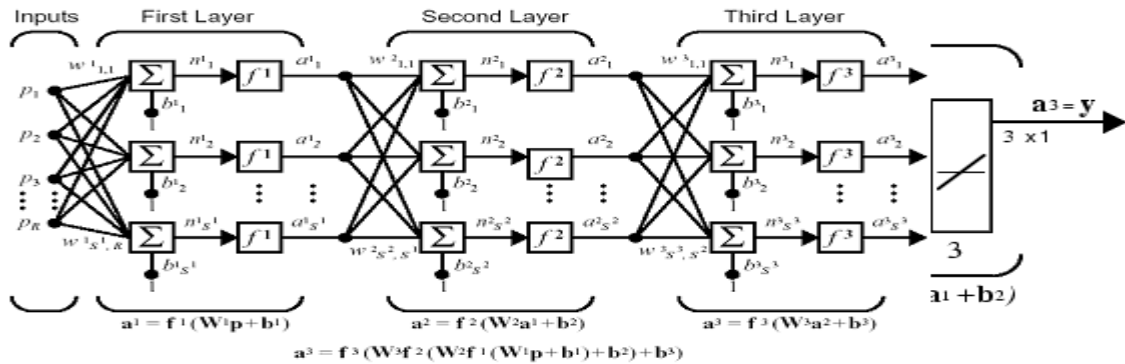


Fig 3.6: Feed forward neural network [69]

hidden layers and the third layer which gives the final output of the entire neural network is called the output layer. We used the number of the layer as a superscript for the weights, neurons, biases, net inputs and outputs from each layer. As shown in Figure 3.6 there are s^1 neurons in the first layer, s^2 neurons in

the second layer, and s^3 neurons in the third layer. There are R inputs to the network and the weight matrix for the first, second, and third layer are represented as w^1 , w^2 , and w^3 .

3.6.2. Working of the feed forward neural network

The ultimate task of the neural network is accomplished in two stages, namely, the training mode and production or testing mode. During the training mode, a set of examples known as the input vectors and their corresponding desired outputs are given to the neural network. The network is trained to learn the relationship that exists between the inputs and the outputs by using a learning algorithm known as the back propagation algorithm. During the training mode, the network, especially the hidden layer neurons, learn to respond to features and gradually the network develops the ability to generalize. After the network has been trained successfully, the next step is to test the neural network by giving it a set of input vectors that were not included in the sets used for training the network. If the network has been trained properly, it should be able to predict the outputs correctly for the input test data set that were never used during the training mode.

In the training mode, the inputs are first passed to the neurons in the first layer. The final output from the neural network is compared with known desired output and error between the actual output of the network and the desired output is determined. The errors are used for adjusting the connection weights associated with the different layers of the network so that the error is progressively decreased in the subsequent prediction of the network. The

training process is repeated until the error has been reduced to a minimum value indicating that the network has learned to the maximum extent possible.

After the training session, the network is put into the testing mode. During testing, the neural network is supplied with a set of test data which was not used during the training session, the neural network would not be given the corresponding outputs associated with the test data inputs. The neural networks response or the output is monitored and compared with the known desired outputs. If the output predicted by the neural network agrees closely with the known desired output with minimum error, it indicates that the network has been trained properly and has attained the power to generalize. During the testing process, the connection weights associated with the various layers of the neural network remain unmodified.

3.7. Learning Algorithms

3.7.1. Least-Mean-Squared (LMS) rule

The least-mean-squared (LMS) algorithm is a kind of supervised training algorithm where the neural network is provided with a set of inputs and their corresponding outputs during training. The algorithm works in such a manner as to reduce the mean square error between the actual network output and target output by adjusting the connection weights and biases of the neurons in different layers of the network. The least mean square error is calculated as follows

$$F(x) = E \left[(t(k) - a(k))^2 \right], \quad (3.3)$$

Where the target output $t(k)$ is the desired output, $a(k)$ is the network output and E represents least mean squared error of the output that has to be reduced. The

least mean squared (LMS) algorithm is also known as Widrow-Hoff algorithm or the delta rule.

The mean square error $F(x)$ can be approximated by

$$\hat{F}(x) = e^2(k) = (t(k) - a(k))^2, \quad (3.4)$$

where k represents the iteration number. The estimate of the gradient is given by

$$\nabla F(x) \cong \nabla e^2(k). \quad (3.5)$$

The partial derivative of $e(k)$ and $e^2(k)$ with respect to the weights for the k^{th} iteration is given by

$$\frac{\partial e(k)}{\partial w_{1,j}} = \frac{\partial}{\partial w_{1,j}} \left[t(k) - \left(\sum_{i=1}^R w_{1,i} p_i(k) + b \right) \right] = -p_j(k), \quad (3.6)$$

$$\frac{\partial e^2(k)}{\partial w_{1,j}} = 2e(k) \frac{\partial e(k)}{\partial w_{1,j}}, \quad (3.7)$$

and the partial derivatives of $e^2(k)$ with respect to the biases at k^{th} iteration is given by

$$\frac{\partial e^2(k)}{\partial b} = 2e(k) \frac{\partial e(k)}{\partial b}, \quad (3.8)$$

Similarly, the derivative of $e(k)$ with respect to the bias is given by

$$\frac{\partial e(k)}{\partial b} = -1. \quad (3.9)$$

Substituting Eqns. (3.6) and (3.9) in Eqns. (3.7) and (3.8), respectively yields the gradient of the squared error for the k^{th} iteration:

$$\nabla F(x) = \nabla e^2(k) = -2e(k) \begin{bmatrix} p_j(k) \\ 1 \end{bmatrix}. \quad (3.10)$$

The steepest descent algorithm, with constant learning rate is given by

$$x_{k+1} = x_k - \alpha \nabla F(x)_{|x=x_k} \quad (3.11)$$

Substitution of Eqn. (3.10) for $\nabla F(x)$ in the Eqn. (3.11) yields

$$x_{k+1} = x_k + 2\alpha e(k) z(k) \quad (3.12)$$

or

$$w(k+1) = w(k) + 2\alpha e(k) p(k) \quad (3.13)$$

and

$$b(k+1) = b(k) + 2\alpha e(k) \quad (3.14)$$

Eqns. (3.13) and (3.14) represent the least mean square algorithm for a single output network. For network with multiple neurons in the outer layer, the LMS algorithm can be written in matrix form as:

$$W_{(k+1)} = W_{(k)} + 2\alpha e(k) p_{(k)}^T \quad \text{and} \quad (3.15)$$

$$b_{(k+1)} = b_{(k)} + 2\alpha e(k) \quad (3.16)$$

3.7.2. Back-propagation algorithm

The back-propagation algorithm is a generalization of the least mean square algorithm. It employs a generalized delta rule. Most of the multilayered feed forward networks, including the one used in this investigation employs back-propagation algorithm. At the end of forward propagation step, the error between the actual network output and targeted output is calculated and based on this error the weights associated with each neuron in the output layer is changed. In back propagation, the network weights are moved along the negative of the gradient of the performance function. The algorithm is known as back propagation algorithm because the change of weights starts from the output layer and then proceeds backwards until all weights associated with the first layer

has been changed in such a way that the error decreases in subsequent predictions of the network.

The mean squared error (MSE) in a multilayered network is given by

$$F(x) = E[e^T e] = E[(t - a)^T (t - a)] \cong e^T(k) e(k) \quad . \quad (3.17)$$

The error in multilayered networks is an implicit function of connection weights of the hidden layers. Therefore, chain rule has to be used for calculating the gradient for the steepest descent algorithm.

The approximate MSE is given by

$$w_{i,j}^m(k+1) = w_{i,j}^m(k) - \alpha \frac{\partial F}{\partial w_{i,j}^m} \quad , \quad (3.18)$$

$$b_i^m(k+1) = b_i^m(k) - \alpha \frac{\partial F}{\partial b_i^m} \quad .$$

Using the chain rule, we get

$$\frac{\partial F}{\partial w_{i,j}^m} = \frac{\partial F}{\partial n_i^m} * \frac{\partial n_i^m}{\partial w_{i,j}^m} \quad (3.19)$$

$$\frac{\partial F}{\partial b_i^m} = \frac{\partial F}{\partial n_i^m} * \frac{\partial n_i^m}{\partial b_i^m}$$

The net input to any layer 'm' is a direct function of the weights and bias in that layer. So it is relatively easy to compute the second terms in the above equations. The net input n_i^m to the layer m is given by

$$n_i^m = \sum_{j=1}^{S^{m-1}} w_{i,j}^m a_j^{m-1} + b_i^m$$

(3.20)

and

$$\frac{\partial n_i^m}{\partial w_{i,j}^m} = a_j^{m-1}, \quad \frac{\partial n_i^m}{\partial b_i^m} = 1$$

The change in the function F with respect to the change in the i^{th} element of the net input at layer m is given by

$$s_i^m = \frac{\partial F}{\partial n_i^m} \quad . \quad (3.21)$$

Using Eqn. (3.21) for sensitivity in Eqn. (3.19), yields

$$\frac{\partial F}{\partial w_{i,j}^m} = s_i^m a_j^{m-1} \quad (3.22)$$

$$\frac{\partial F}{\partial b_i^m} = s_i^m$$

Now the steepest descent algorithm can be conveniently expressed in the following form the using matrix notation

$$\begin{aligned} W^m(k+1) &= W^m(k) - \alpha s^m (a^{m-1})^T \\ b^m(k+1) &= b^m(k) - \alpha s^m \end{aligned} \quad , \quad (3.23)$$

$$s^m = \frac{\partial F}{\partial n^m} = \begin{bmatrix} \frac{\partial F}{\partial n_1^m} \\ \frac{\partial F}{\partial n_2^m} \\ \vdots \\ \frac{\partial F}{\partial n_{s^m}^m} \end{bmatrix} \quad , \quad (3.24)$$

where W^m , b^m and s^m are the weight matrix, bias vector, and the sensitivity vector, respectively.

As the term back propagation indicates, in this algorithm the sensitivity at layer m is computed using the sensitivity at its succeeding layer $m+1$. The Jacobian matrix is given by

$$\frac{\partial n^{m+1}}{\partial n^m} = \begin{bmatrix} \frac{\partial n_1^{m+1}}{\partial n_1^m} & \frac{\partial n_1^{m+1}}{\partial n_2^m} & \dots & \frac{\partial n_1^{m+1}}{\partial n_{s^m}^m} \\ \frac{\partial n_2^{m+1}}{\partial n_1^m} & \frac{\partial n_2^{m+1}}{\partial n_2^m} & \dots & \frac{\partial n_2^{m+1}}{\partial n_{s^m}^m} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial n_{s^{m+1}}^{m+1}}{\partial n_1^m} & \frac{\partial n_{s^{m+1}}^{m+1}}{\partial n_2^m} & \dots & \frac{\partial n_{s^{m+1}}^{m+1}}{\partial n_{s^m}^m} \end{bmatrix} . \quad (3.25)$$

Now,

$$\begin{aligned} \frac{\partial n_i^{m+1}}{\partial n_j^m} &= \frac{\partial \left(\sum_{l=1}^{s^m} w_{i,l}^{m+1} a_l^m + b_i^{m+1} \right)}{\partial n_j^m} , \\ &= w_{i,j}^{m+1} \frac{\partial a_j^m}{\partial n_j^m} = w_{i,j}^{m+1} \frac{\partial f^m(n_j^m)}{\partial n_j^m} = w_{i,j}^{m+1} \dot{f}(n_j^m) \end{aligned} \quad (3.26)$$

using Eqn. (3.26), the Jacobian matrix can be rewritten as

$$\frac{\partial n^{m+1}}{\partial n^m} = W^{m+1} \dot{F}^m(n^m) \quad (3.27)$$

$$\dot{F}^m(n^m) = \begin{bmatrix} \dot{f}(n_1^m) & 0 & \dots & 0 \\ 0 & \dot{f}(n_2^m) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \dot{f}(n_{s^m}^m) \end{bmatrix}$$

Now using the chain rule again, the recurrence relation can be expressed in matrix form as

$$s^m = \dot{F}^m(n^m) (W^{m+1})^T s^{m+1} , \quad (3.28)$$

$$s^M \rightarrow s^{M-1} \rightarrow s^3 \rightarrow s^1 .$$

From the above expression, we notice that the sensitivity is propagated backward from the last layer to the first layer in the network. The sensitivities at the output layer is expressed in matrix form as

$$s^M = -2 \dot{F}^M(n^M)(t - a) \quad . \quad (3.29)$$

3.7.3 Levenberg-Marquardt algorithm

The basic back propagation algorithm is often the simplest and slowest minimization method. The Levenberg-Marquardt algorithm provides faster convergence and is successfully used to speed up the convergence of back propagation. It should be noted that Levenberg-Marquardt algorithm uses the backpropagation procedure in which derivatives are processed from the last layer of the network to the first. Hence, the Levenberg-Marquardt algorithm could be called a backpropagation algorithm.

The second-order Taylor series is represented as follows

$$F(w + \Delta w) = F(w) + g^T \cdot \Delta w + \frac{1}{2} \Delta w^T \cdot H \cdot \Delta w + \dots ,$$

where Δw is the adjustment to the weight, g is the gradient vector and H is the Hessian matrix. They are defined as:

$$g = \frac{\partial F(w)}{\partial w}$$

$$H = \frac{\partial^2 F(w)}{\partial w^2} ,$$

$$\Delta w = -H^{-1} \cdot g$$

In quasi Newton method, Hessian matrix is estimated by some positive definite matrix, which ensures the convergence. The Hessian matrix is approximated as:

$$H \cong 2J^T J ,$$

where J is the Jacobian matrix that contains the first derivatives of the network errors with respect to the weights and biases.

Levenberg-Marquardt algorithm uses an approximation to the Hessian matrix and is given by

$$\Delta w = [J^T J + \mu I]^{-1} J^T ,$$

where J is the Jacobian matrix and μ is a scalar.

CHAPTER 4

LITERATURE REVIEW

Diamond has a unique combination of physical, chemical, electrical, and optical properties which make it a potential candidate for numerous industrial applications. For example, its very high hardness and wear resistance make it ideal for cutting tools and grinding wheels; its insulating properties, radiation hardness and high thermal conductivity make it an ideal member for applications in circuit packaging, high power, electro-optic, semiconductor devices and optical devices [20]. In the 80's diamond films were grown at low pressures under metastable conditions using chemical vapor deposition (CVD) techniques, such as hot filament CVD, microwave plasma assisted CVD and DC arc plasma jet and flame plasma deposition [21-23]. Due to the limited growth rate ($<0.1 \mu\text{m/hr}$), CVD process cannot effectively compete with the HP-HT process on the basis of growth rate or overall cost. However, the low pressure diamond synthesis has led to a new era in diamond technology. (Extensive review presented by DeVries [24]). There are, however, some applications where LP-CVD diamond synthesis is preferred over HP-HT synthesis. For example, the low-pressure CVD process has a great potential for optical, infrared, and X-ray applications as well as for a number of manufacturing and tribological applications. For example, in the manufacturing area, diamond coatings on cutting tools by the CVD technique can

be used to improve wear resistance, thereby improving the tool life. The diamond crystals obtained through the CVD technique find applications as cutting tools in nanometric machining and grinding operations and as heat sinks in electronic applications.

Recently, nanocrystalline CVD diamond films have been synthesized with superior properties, such as, higher toughness, lower light scattering and higher Young's modulus [25]. Nanocrystalline diamond (NCD) films are composed of diamond grains of the order of 50 nm, and display under certain conditions smooth morphology. They have been considered for applications in micro-electro-mechanical systems (MEMS) and its nano variant, namely, nano-electro-mechanical-systems (NEMS), where the mechanical, electrical, and corrosion properties of these fully dense films extend the range of applications of these novel devices [26]. Gruen *et al.* [27-31] have conducted extensive research on nanocrystalline diamond film.

One of the key factors in the CVD diamond film growth is the nature of the hydrocarbon species used. Gruen *et al.* [27] reported successful growth of diamond films using fullerene precursors in an argon microwave plasma without the addition of hydrogen or oxygen. The average grain size of the films obtained is reported to be 0.05 μm . They postulated that collisional fragmentation of C_{60} to give C_2 could be responsible for the high growth rate of the very-fine-grained diamond films. Zhou *et al.* [28] investigated the transition from microcrystalline to nanocrystalline films grown from $\text{Ar}/\text{H}_2/\text{CH}_4$ microwave plasma; the transition becomes pronounced at an Ar/H_2 volume ratio of 4, and the microcrystalline

diamond films are totally transformed to nanocrystalline at an Ar/H₂ volume ratio of 9. They suggested that the transition in the microstructure to be due to change in the growth mechanism from CH₃ in high hydrogen content to C₂ as the growth species in low hydrogen content plasmas.

Goyette *et al.* [29] experimentally determined the density of gas phase C₂ in Ar-H₂-CH₄ and Ar-H₂-C₆₀ plasmas and reported C₂ to be an important species in these growth environments.

Gruen *et al.* [30] used optical spectroscopy to examine C₆₀/Ar plasma and noticed the spectrum to be dominated by swan bands of C₂. They proposed that collisionally induced dissociation of C₆₀ in argon plasmas could be the mechanism for C₂ production and that C₂ is the principal growth species in their diamond film growth experiments.

The above studies indicate that carbon dimer (C₂) is an important growth species for nanocrystalline diamond growth and the elementary reactions of carbon dimer on a diamond substrate surface is of interest.

A variety of hydrocarbon species are used in the microwave plasma CVD process for diamond growth. Some of the widely employed hydrocarbon species are C₂H₂, C₂H, CH₃, C₂, C, and C₂H₄. The mechanism by which diamond film growth occurs differs from one hydrocarbon species to another and it also plays a vital role in the properties of the films obtained. Hence, the investigation of the mechanisms by which diamond film growth takes place and various reactions that occur between the gaseous hydrocarbon molecules and the substrate on which the films are grown is always of considerable interest.

Any complex reaction between the gaseous precursors and the atoms of the substrate involve basic elementary events, such as chemisorption, insertion, scattering, and desorption. The probability of occurrence of each of these events is greatly affected by factors, such as the incident translational energy, rotational energy of the molecules, the angle of incidence, rotation angle, temperature of the substrate, and impact parameter. Hence, the investigation of the effect of these parameters on various event probabilities in CVD and other film growth processes has attracted considerable attention among research community.

Ulloa *et al.* [31] investigated the adsorption of hydrocarbons, such as CH_3 , CH_2 , and C_2H_4 on the flat terraces and near step edges of diamond (100) surfaces using MD simulations. They found that adsorption of CH_3 on the (100) face and subsequent abstraction of one of its hydrogen will promote β - scission essential for continued growth. Alfonso and Ulloa [32] studied methyl radical deposition on diamond (100) surfaces using MD simulations. The time step employed in their simulation was between 0.25 – 0.5 fs and most of their trajectories were monitored until elapse time of 2.5 ps. They reported that the adsorption probability of the CH_3 radical on diamond substrate increases with the kinetic energy of the methyl radical and decreases with the incidence angle but the rise in adsorption probability becomes less pronounced as kinetic energy of the incident CH_3 goes up. Hydrogen knock out events were reported to occur when the methyl radical is incident with a normal energy above 1 eV and is found to be more pronounced for a normal energy of 10 eV.

Hu and Sinnott [33] examined the deposition of an ethylene molecular-cluster beam at various incident angles and incident energies on a diamond (111) surface that was terminated at the top and bottom with hydrogen atoms using a reactive potential coupled to Lennard-Jones (L-J) potential. The substrate had 24 layers of carbon atoms and it contained 13700-13900 atoms with an impact plane area of $69 \times 40 \text{ \AA}^2$. All simulations were carried out for 3 ps and the time-step used was 0.2 fs. They predicted that with an increase in the incidence angle, the amount of adhesion of a thin film decreases. They also reported that crystallographic orientation and the incidence angle have less effect on the film structure and formation.

Wang *et al.* [34] investigated the deposition of CH_3 and CH_2 radicals on diamond (001) surfaces at room temperature to determine the energy threshold (E_{th}) for chemisorption and reported that for CH_3 , the value of E_{th} on diamond (001)-(2x1) H surface is higher than that on diamond (001)-(2x1) surface and lower than that of C_2H_2 on the diamond (001)-(2x1) surface.

Perry and Raff [35, 36] computed rate coefficients, event probabilities, and desorption probabilities for many elementary chemisorption reactions on a diamond ledge and diamond terrace structures at 1250K. The diamond (111) terrace substrate they used had a total of 145 lattice atoms, a trajectory was carried out between 0.5 - 1.5 ps and each trajectory took an average of ~ 30 min of CPU time on a Digital (DEC ALPHA 3000/Model 400) workstation. The diamond ledge surface had a total of 147 atoms, the individual trajectory time varied over the range 0.1-0.87 ps, and the CPU time was the same as for the

terrace structure. They investigated molecules and radicals, such as C_2H_2 , C_2H , CH_3 , CH_2 , C , C_2 , C_3H and found that chemisorption rate is lower for nonradical species, such as C_2H_2 and C_2H_4 than for radicals (10^{12} - 10^{13} $cm^3/mols$). They reported that CH_3 is the least reactive and atomic carbon has the largest chemisorption rate of all the species investigated.

Izumi *et al.* [37] investigated the reaction probability of silane molecules on silicon (001) surface. They considered twenty different substrate conditions to overcome the scattering effect due to vibration of the substrate. They carried out a total of 20000 trials to obtain probability on the order of 10^{-3} . They reported that the reaction probability depends significantly more on the internal energy of the silane than on the substrate temperature. The reaction probability increased linearly with the translational energy; quadratically with the vibrational energy of the silane molecule; and depends less on its rotational energy.

Zhu *et al.* [38] studied the interaction between low energy C_2H_2 and diamond (001)-2x1 surface, 200 trajectories were considered and each trajectory lasts for around 3 ps. Six types of chemisorption configurations (S1 through S6) were noticed as shown in Figure 4.1. The S1 structure is found to be the most stable because of its high binding energy and S6 structure to be the least stable.

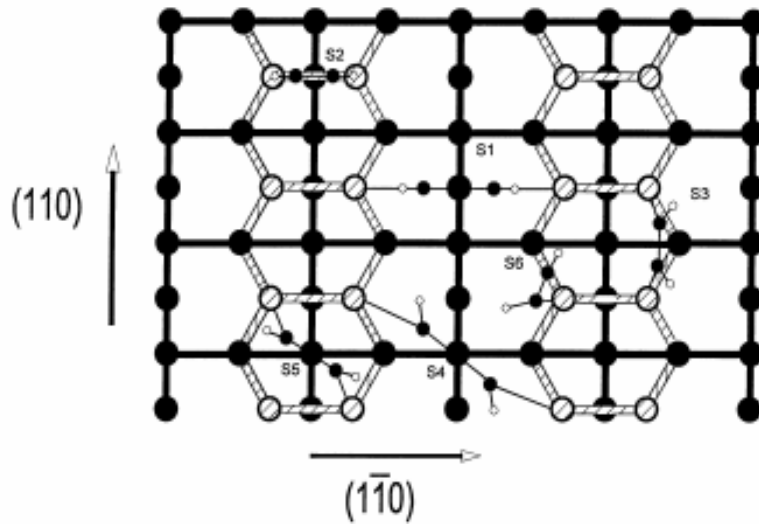


Fig. 4.1: Six types of chemisorption configurations on diamond (001) – (2 x 1) surface [38].

The S2 and S3 configurations are found to be most frequently occurring configurations during diamond film growth and thereby playing an important role in diamond synthesis.

Hansen and Hudson [39] studied interaction of oxygen molecules with clean and oxygen covered Ge (100) surfaces using molecular beam scattering techniques. They reported that sticking coefficient increased from 0.018 ± 0.002 to 0.079 as the incident beam energy was increased from 2.1 to 7.9 kcal/mol, but it decreased from 0.0176 at normal incidence to a minimum value when the incident angle of the beam is increased to $\theta = 70^\circ$.

Belsky *et al.* [40] used MD simulations to study the sticking probability of Cu and Ta atoms on FCC Cu (111) and BCC Ta (110) crystal faces, respectively, for different values of incident energies ranging from 0 to 150 eV and angle of incidence from 0° to 90° . The time step for the trajectories was on the order of 0.1 – 1 fs and the total duration for every trajectory was on the order of 100 – 1000

fs. The sticking probability of both Cu and Ta atoms was found to be inversely proportional to the incident beam energies. The sticking probability was found to decrease first with the incidence angle, but at larger angles, the probability began to increase. This behavior was ascribed to increased interaction time of the incoming atom with the substrate and a low normal velocity component. Vattuone *et al.* [41] investigated chemisorption of O₂ on Ag (001) surfaces at 100 K by reflectivity method. The sticking probability was found to increase monotonously until the translational energy of the O₂ molecule reached 0.7 eV and decreased thereafter because at high energies the collided molecules still retain sufficient energy to avoid being trapped on the surface and are able to escape into the gas phase by scattering inelastically off the repulsive part of the chemisorption potential.

Huang *et al.* [42] studied the interaction between low-energy CH₃ and diamond (001)-(2x1) at room temperature using MD simulations. An energy threshold (E_{th}) of 8 eV below which no chemisorption of CH₃ would occur was noticed. Hydrogen dissociation from CH₃ was observed for incident energies higher than 15 eV. They also found that below 10 eV incident energy, the chemisorption probability of C₂H₂ on a clean diamond (001)-(2x1) surface was lower than that of CH₃ on a hydrogen covered surface at the same impact energy.

Neyts *et al.* [43] investigated the sticking efficiencies and hydrogen abstraction efficiencies for hydrocarbon species, such as C₂, C₂H, C₃H₂, and C₃ on a diamond like carbon (DLC) layer at two different values of their initial kinetic

energies, namely, 0.1 eV and 1.0 eV using MD simulations. The DLC layer had 830 atoms, the time step used was 0.5 fs, and the trajectory duration was 1.25 ps for 1 eV kinetic energy impacts and 2.5 ps for 0.1 eV kinetic energy impacts. All species had sticking efficiencies between 0.1 and 0.4. They found that species with no hydrogen atom had a smaller decrease in sticking efficiency with decreasing energy than species that contain hydrogen. They also noted that C_3H_2 had the highest hydrogen abstraction efficiency, C_2 had the lowest abstraction efficiency, and C_3 had zero hydrogen abstraction efficiency.

Palithorpe [44] conducted MD simulation studies for the deposition of low-energy carbon atoms onto a low-temperature diamond (111) surface using Stillinger-Weber potential. The time step for the integration was 0.13 fs, the energy of the incident carbon atom was in the range of 1-100 eV and the substrate temperature was maintained at 100 K. He reported that with intermediate energies (20-60 eV), the incident atom penetrates beneath the exposed (111) surface and significantly increases the lateral compressive stress in the diamond film, thereby promoting amorphous diamond formation.

From the above literature survey, we infer that MD simulation is a powerful tool for studying many complex reactions that takes place when different species are incident on the diamond substrate. But a main disadvantage of MD simulation is that it consumes huge amounts of computational time (and consequently higher cost), and this time increases rapidly as the number of atoms in the system considered increases. Even to study the effect of one parameter, say the effect of impact parameter on the chemisorption probability of

a carbon dimer on (100) diamond surface, keeping other inputs at a fixed value, consumes large amounts of computational time. Therefore, the exploration of the entire set of values become highly challenging and computationally costly when MD simulation has to be used. Any technique that permits the exploration of the wide range of input conditions will be of great benefit this area. Neural network is one such approach that offers such an advantage. This can be inferred from the following literature.

Natale *et al.* [45] employed a modular neural network approach for enhancing the quality of films obtained during atmospheric chemical vapor deposition of doped silicon dioxide films. A neural network was used to establish a relationship between the equipment's operating conditions and the characteristics of the resulting films. This in turn aids in finding the optimal set up conditions for obtaining high quality of films. Machine operating conditions are determined by factors, such as gas flow, chamber pressure, injector temperature, nitrogen flow. These factors are used as inputs to the network and film quality deciding factors, such as boron and phosphorus weight percent in the plasma and film thickness were used as outputs to the network for training and testing the neural network. The prediction by the network was good with an average error of about 1 % and a maximum error below 10 %. Erbil *et al.* [46] developed a semi-empirical model using hybrid neural networks to determine the deposition rates of TiO_2 films in a metal-organic CVD process. Temperature, total flow rate, reactor chamber pressure, source pressure, and precursor flow rate were used as inputs to the network and the TiO_2 deposition rate was used as output during

the learning and testing stages of the network. The neural network used was able to identify three critical parameters, unknown in analytically derived deposition rate expression, leading to more general physical expression and methodology for predicting deposition rate over a wide range of operating conditions.

Bhatikar and Mahajan [47] used a feed-forward neural network to predict the performance of a CVD barrel reactor widely used in silicon epitaxy. Their approach involved spatial variation of the deposition rate of silicon on a facet of the reactor. They hypothesized that this spatial variation encodes a pattern that reflects the state of the reactor. A feed-forward neural network with eight neurons in the hidden layers was used to predict and decode the pattern thereby predicting the state of the reactor so that it can be optimized to increase the production efficiency. Three different patterns or process faults were diagnosed and the network was able to predict and discriminate these process faults with 100% accuracy.

Han and May [48] applied neural networks to predict the complex correlation between the deposition conditions and output parameters reflecting film quality in plasma enhanced CVD (PECVD) process. Deposition parameters, such as, substrate temperature, pressure, RF power, silane flow, and nitrous oxide flow were used as inputs and the corresponding deposition rate, permittivity, film stress, uniformity, silanol and water concentration in the films deposited were used as outputs for training the network. This trained network model was used “in reverse” to predict the necessary operating conditions to achieve the desired film quality. They were able to synthesize recipes to produce

novel film properties, such as uniformity, low permittivity, stresses, and impurity concentration using the optimized neural network models.

Geisler *et al.* [49] modeled chemical vapor deposition of silicon nitride (Si_3N_4) films using a five-layered feed forward network. The neural network was trained using both supervised and unsupervised learning techniques. The input vector had six variables, namely, substrate temperature, chamber pressure, RF power, NH_3 flow, SiH_4 , and N_2 flow and the output vector had three variables, namely, the film's refractive index, the effective lifetime, and the positive charge density. Competitive learning algorithms were then used to determine the input-output relationship functions and these functions are then optimally adjusted to enhance the silicon nitride film properties.

Lorenz *et al.* [50] used a multilayered feed-forward network and developed an *ab-initio* potential energy surface (PES). They showed the accuracy of the neural network developed PES using the hydrogen dissociation on the (2 x 2) potassium covered Pd (100) surface. The sticking probability of H_2 on this potassium (2 x 2) covered Pd (100) surface is calculated using MD simulations on the neural network PES. The results were compared with the analytically developed potential energy surfaces and found to be in good agreement.

Hobday *et al.* [51] showed that a feed-forward network can be used to develop a potential energy surface to study the complex C-H problem. The network used had an input vector set with five elements and six hidden nodes with a total of 43 weights and biases. The results were compared with the Brenner potential formulation for C-H clusters which indicated a good agreement

with both structure and energetics. Numerical experiments showed that the PES developed using neural network though slower than Brenner potential by 60-80 % it is still inexpensive compared to the *ab initio* calculations and can be efficiently used for still more complex systems like C-N where bonds are more complex as compared to C-H systems considered.

Raff *et al.* [52] interpolated *ab initio* potential-energy surfaces using a feed forward neural network and novelty sampling approach. They used various configurations of five-atom silicon cluster and calculated the force and potential associated with each configuration at the MP4(SDQ) level of accuracy using 6-31G** basis set. They employed a novel sampling procedure and sampled the important regions of configuration space in iterative fashion using MD trajectories. A large number of new cluster configurations and corresponding potential and forces associated with those configurations were obtained using the novelty sampling technique. These cluster configurations, and the potential and forces associated with them were used to fit a neural network and obtain the potential energy surface (PES). The interpolated potential energy surface (PES) can be used efficiently for conducting MD and Monte Carlo studies of large systems involving complex reactions, nanometric cutting and nanotribology. The novelty sampling technique involves tight integration of MD calculations with NN and enables easy identification of new configurations in MD and also act as a good convergence test independent of MD computations. Early stopping and regularization techniques were used to give quick and precise results. The neural

network used was found to give good interpolation accuracy and easy usage of the obtained force fields directly for dynamic studies.

Sumpter and Noid [53] employed a neural network with 426 input nodes, one hidden layer with 7 nodes, and an output layer of 18 nodes for obtaining a potential energy surface for macromolecules, such as a polyethylene molecule. An accurate anharmonic potential energy surface was formulated. The parameters in this PES were suitably changed and the corresponding vibrational spectra of the macromolecule is monitored. The neural network is then trained for 51 different vibrational spectra values of the macromolecules as inputs and the corresponding potential energy parameters outputs for 20000 cycles. Then the network was trained to determine the relation between the vibrational spectra and the corresponding parameters of the PES with a maximum error of less than 4%. This network was later used for obtaining parameters for a multidimensional PES.

Noid *et al.* [54] used neural network to investigate the energy flow in molecular systems, such as H_2O_2 . The neural network was made to learn the correlation between phase-space points along a classical trajectory and mode energies for stretch, bend, and torsion vibrations. The input vector to the network comprises of 12 cartesian atom positions (x, y, z), 12 cartesian momenta (p_x, p_y, p_z), and four atomic masses. The output from the network comprised of six kinetic internal mode energies. The network employed had 28 input nodes, two hidden layers with 38 nodes in the first hidden layer and 12 nodes in the second hidden layer and an output layer of six nodes, giving a total of 84 nodes and

1648 connection values including bias values. The trained network was able to produce reasonably accurate results with an average error between 1% and 12%. Also the network has been employed for studying the energy flow in other tetratomic molecules, such as H_2X_2 , $\text{X}=\text{C}$, and Se.

From the above review of literature on neural networks, we find the application of neural networks to molecular dynamics can reduce the burden on MD simulations to a considerable extent. It can also provide an opportunity to explore the effect of different parameters on the probabilities of various events in a CVD process with less computational time and cost.

CHAPTER 5

PROBLEM STATEMENT

From a review of the literature one can perceive that investigation of elementary reactions, such as chemisorption, scattering, insertion, and desorption that occur during thin-film growth of microcrystalline diamond by chemical vapor deposition (CVD) process or any other growth process is of immense importance. Though MD simulations have been used successfully to study these reactions, a major limitation inherent with it is that it involves integration of numerous equations that consume an enormous amount of computational time and cost.

For example, to study the influence of one of the input parameters, say the effect of impact parameter (b) on the chemisorption probability for a fixed set of other parameters, namely, translational energy (E_{Trans}), rotational energy (E_{Rot}), incidence angle (θ), and azimuthal angle (Φ) of the carbon dimer, we ran 50 trajectories for every value of impact parameter (b), ranging from $b = 0$ to $b = 3.5$ Å. By dividing this range into ten equal intervals, a total of 550 trajectories were run. We used a system of 324 atoms and time for running a single trajectory was ~1 minute. So, it took us a total of 550 minutes (~9.2 hours) to study the influence of a single parameter on one of the reaction probabilities for a single set of other input parameters.

Therefore, the investigation of the entire domain consisting of the dependence of various reaction probabilities on different variables is a very time consuming and challenging job. Any technique that cut shorts the enormous simulation time and offers an opportunity to explore the entire domain of input variables will be of immense interest to researchers and deserves to be explored. To achieve this objective, we intend to use neural networks to determine the probabilities of various reaction events, namely, chemisorption, scattering, and desorption that occur during the deposition of carbon dimer on a diamond (100) surface as a function of various input parameters.

The implementation of the neural network for predicting the probabilities of various events occurring during carbon dimer (C_2) deposition on a diamond (100) surface is achieved through following stages:

- We first employed MD simulations to compute the probabilities of chemisorption, scattering, and desorption as a function of input parameters, such as rotational energy (E_{Rot}), translational energy (E_{Trans}), angle of incidence (θ), impact parameter (b), and rotation angle (Φ).
- Training the neural network by feeding the values of outputs for some known values of input parameters over a wide range.
- Testing the NN by supplying it with few input parameters and asking it to predict the output, and comparing the NN prediction with MD results.

CHAPTER 6

MOLECULAR DYNAMICS (MD) SIMULATIONS OF CARBON DIMER (C₂) DEPOSITION ON DIAMOND (100) SURFACE

Diamond film growth in a chemical vapor deposition (CVD) process involves complex reaction mechanisms taking place between the atoms of the substrate and the gaseous radicals used in the process. However complex the reaction mechanism might be, it all starts with preliminary elementary reactions, such as chemisorption of the radical species, scattering and desorption. This chapter focuses on these elementary reactions that occur during the deposition of carbon dimer (C₂) on to the (100) diamond surface in a CVD process.

6.1. Computational Model

In this study, a (100) diamond surface has been used. The surface was modeled using a slab of five layers of carbon atoms with the (100) face exposed. Except for one carbon atom that serves as the radical site, every carbon atom on the top layer is capped with one hydrogen atom. The atoms on all five faces of

the substrate, except the top face, are made to be non-moving atoms, while the remaining atoms were allowed to move. The system used in this study has a total of 324 atoms of which two are the atoms of the carbon dimer and remaining 322 atoms are of the diamond substrate and hydrogen atoms. The dimensions of the substrate are $17.7 \times 3.54 \times 17.7$ Å. The two atoms of the carbon dimer are placed above the radical site such that the center of mass of the dimer is at a vertical distance of ~ 10 Å from the top surface of the substrate to make sure that the long-range interactions between the carbon dimer and the substrate atoms is near zero.

An empirical many-body Brenner potential (Brenner *et al.* [13]), which realistically describes the bonding in hydrocarbon systems, is used to account for the short-range interactions. A Lennard – Jones 6-12 potential is used to model the long-range interactions. The substrate temperature was maintained at $T_s=600$ K using a thermostat that employs the velocity scaling method of Berendsen [55]. A constant time step of 0.5 fs is used for numerical integration of the equations of motion and the Gear predictor corrector [18] method was used for numerical integration. Before the dimer deposition process, the substrate is relaxed in a 600 K thermal bath for 30 ps allowing it to approach the thermal equilibrium state. The simulation model used is shown along with the carbon dimer (C_2) in Figure 6.1. The top three layers of the diamond substrate and the radical site are shown in Figure 6.2.

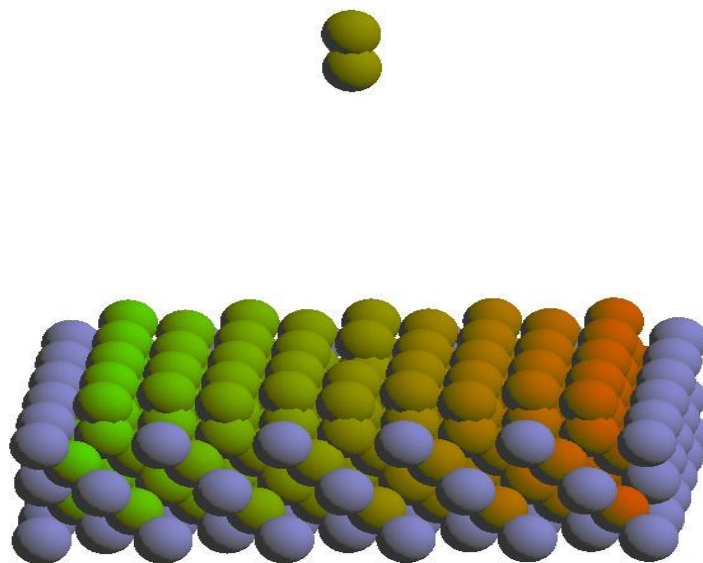


Fig 6.1: Simulation model and carbon dimer (C_2)

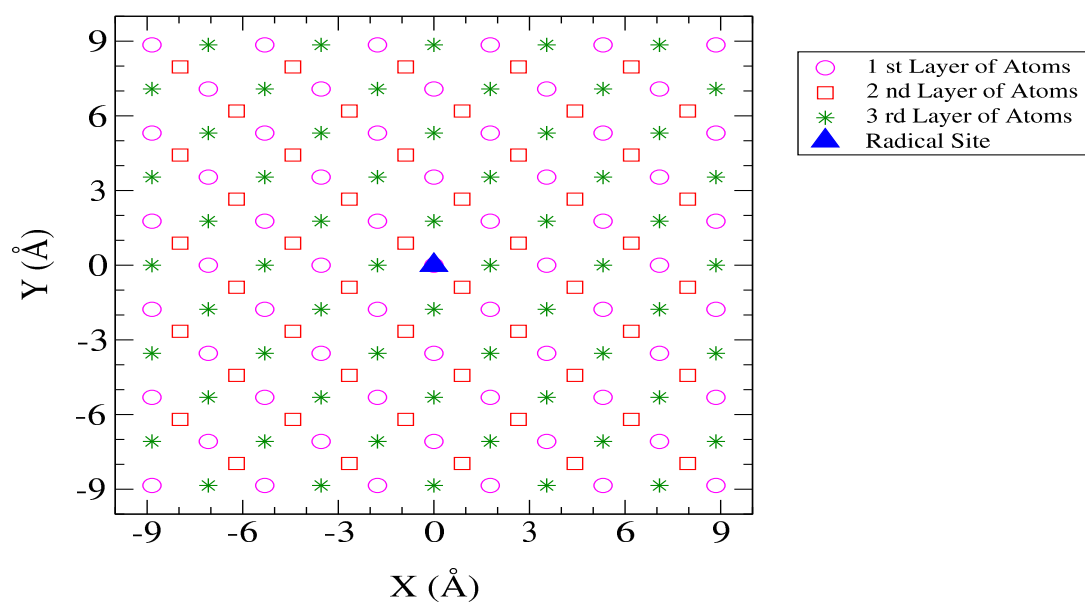


Fig 6.2: Top three layers of atoms of diamond (100) substrate and radical site

6.2. Parameters of Interest

The mechanism of diamond-film growth has been investigated by several research groups using theoretical and/or experimental methods. A number of elementary reactions have been suggested as playing a vital role in diamond-film formation. The occurrence of these reactions depend not only on the surface structure of the substrate on which the hydrocarbon atoms are deposited but also on a number of parameters [31, 33, 35, 36, 38], such as

- Incident azimuthal angle (θ)
- Rotation angle (Φ)
- Impact parameter (b)
- Translational energy of the Carbon dimer (E_{Trans})
- Rotational energy of the Carbon dimer (E_{Rot}).

In the following section we will be dealing with the distribution of each of these input parameters as well as with the events considered in this investigation.

6.2.1. Incident polar angle (θ):

Incident angle of the dimer (θ) is the angle between the velocity vector of the center of mass of the dimer and the normal from the aimed point on the substrate. The polar angles were selected from the distribution function $P(\theta) d\theta = C \sin\theta d\theta$ over the range $0 \leq \theta \leq \theta_{\text{max}}$. This can be conveniently accomplished using a cumulative distribution function that leads to Eqn. (6.1) [56].

$$\Theta = \cos^{-1}\{1 - \xi_1 (1 - \cos \theta_{\text{max}})\} , \quad (6.1)$$

where ξ_1 is a random number selected over the range $[0, 1]$ and θ_{max} is determined as follows. In the case of an infinite lattice model, $\theta_{\text{max}} = \pi/2$. In the

present calculations, the size of the lattice model used requires that the value of θ_{\max} be limited to 23° . With this choice for θ_{\max} , normalization of the distribution function gives $C=12.579$. The incidence angle of the hydrocarbon atom is shown along with other input parameters in Figure 6.3

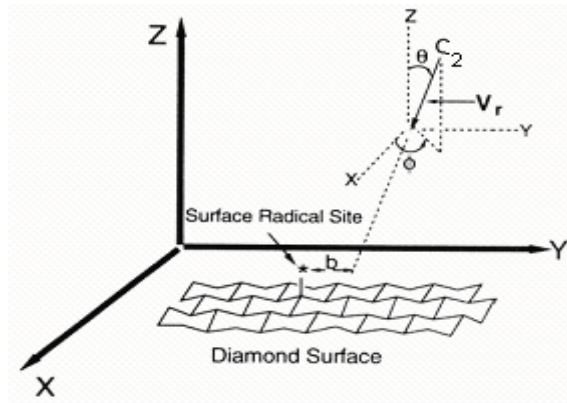


Fig 6.3: Sketch showing input variables for the trajectory calculations [35]

The distribution for the incidence angle of the dimer follows the smooth linearly increasing curve shown in Figure 6.4. The histogram is an example of the θ distribution obtained using Eqn. (6.1)

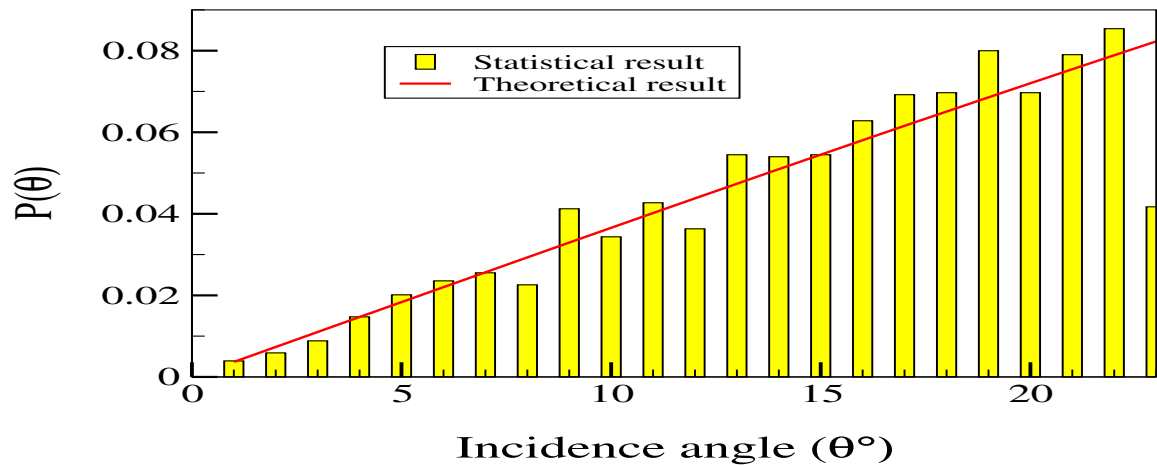


Fig 6.4: Distribution of incidence angle

6.2.2. Rotation angle (Φ):

The probability distribution for the rotation angle is of the function form $P(\Phi) d\Phi = C d\Phi$ and is uniform over the interval $0 \leq \Phi \leq 2\pi$; the initial value of Φ has been selected from in Eqn. (6.2)

$$\Phi = 2 \pi \xi_2 , \quad (6.2)$$

where ξ_2 is a random number selected over the range $[0, 1]$. The normalization of the distribution function gives $C = 0.1591$. The theoretical distribution of the rotation angle is a constant straight line shown in Figure 6.5 as the line. The statistical result obtained from Eqn. (6.2) [56] is shown as the histogram.

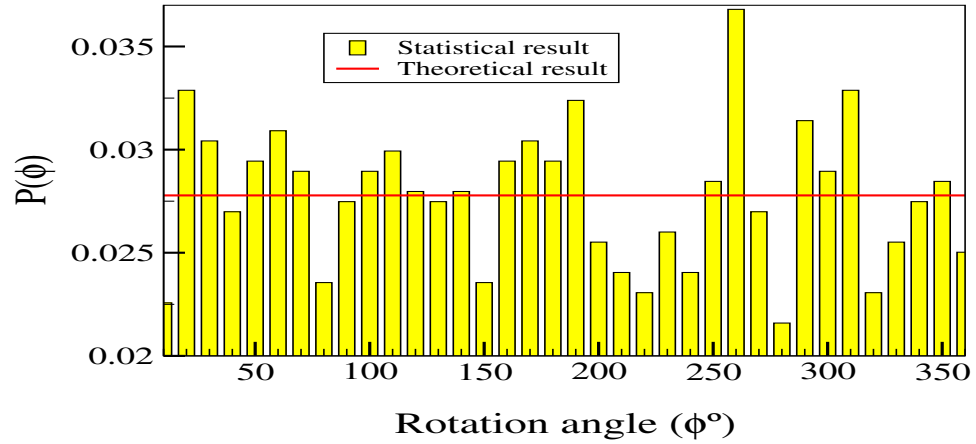


Fig 6.5: Distribution of rotation angle

6.2.3. Impact parameter (b):

The impact parameter represents the distance between the radical site and the aiming point on the surface of the substrate. The impact parameters are

selected from the distribution $P(b)db = N2\pi bdb$ over the range $0 \leq b \leq b_{\max}$, where the upper limit, b_{\max} , is chosen such that for impact parameters $b > b_{\max}$, the chemisorption probability is zero. Using a cumulative distribution function, this selection can be made by obtaining b for each trajectory from the equation

$$b = \sqrt{\xi_3} b_{\max} , \quad (6.3)$$

where ξ_3 is a random number selected from a uniform distribution on the interval $[0, 1]$. The maximum impact parameter (b_{\max}) is found to be 3.5 Å. With this choice for b_{\max} the normalization of the distribution function gives $N=0.1633$. The impact parameter distribution obtained using Eqn. (6.3) is compared with the theoretical result obtained from the probability distribution function [36].

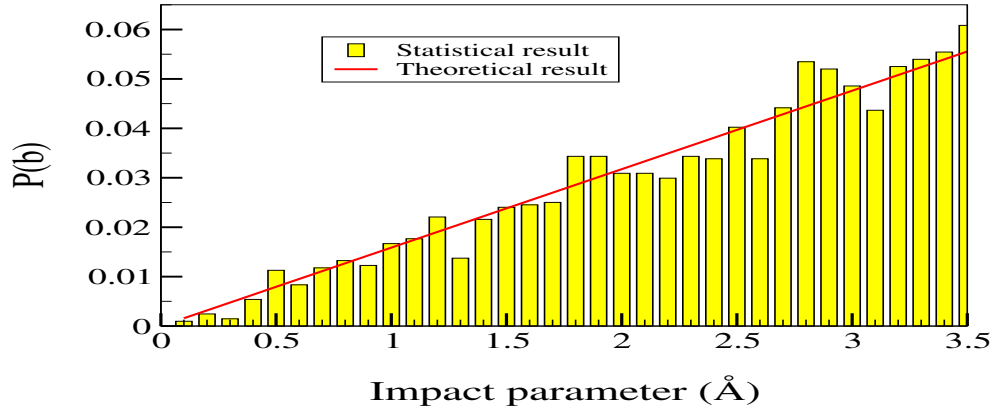


Fig 6.6: Distribution of impact parameter

6.2.4. Translational energy of the carbon dimer (E_{Trans})

The initial translational velocity of the carbon dimer was selected from a Boltzmann distribution at the same temperature as the lattice which is $T_s = 600$ K. The functional form of the Boltzmann distribution is given by [57]

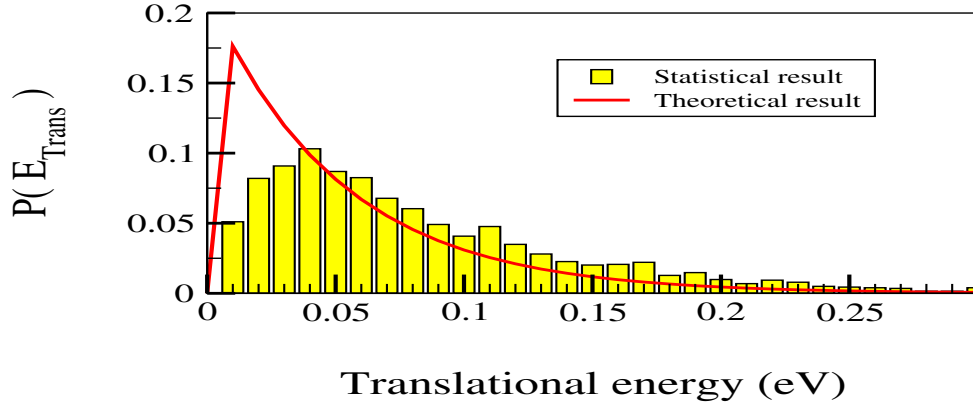


Fig 6.7: Distribution of translational energy of the dimer

$$P(E_{\text{Trans}}) = A e^{-E/KT}, \quad (6.4)$$

The distribution of translational energy of the dimer is shown in Figure 6.7.

6.2.5. Rotational energy of the carbon dimer (E_{Rot})

The rotational energy of the carbon dimer was calculated assuming a rigid-rotor type rotational energy quantization [58]:

$$E_J = J(J+1)\hbar^2 / 2I, \quad \text{where } \hbar = h / 2\pi. \quad (6.5)$$

Here, I represents the equilibrium moment of inertia and J represents a continuous quantum number [58] given by the Eqn (6.6)

$$J = 1/2 \left(\left\{ 1 - 8IkT \ln(1 - \xi) / \hbar^2 \right\}^{1/2} - 1 \right), \quad (6.6)$$

where T is the temperature and ξ is a random number selected from a uniform distribution in the interval $[0, 1]$. The spread for the rotational energy of the dimer is shown in Figure 6.8. The theoretical result for the rotational energy is given by the distribution function [58] in Eqn (6.7).

$$P(J) dJ = C g_J \exp(E_J / kT), \quad (6.7)$$

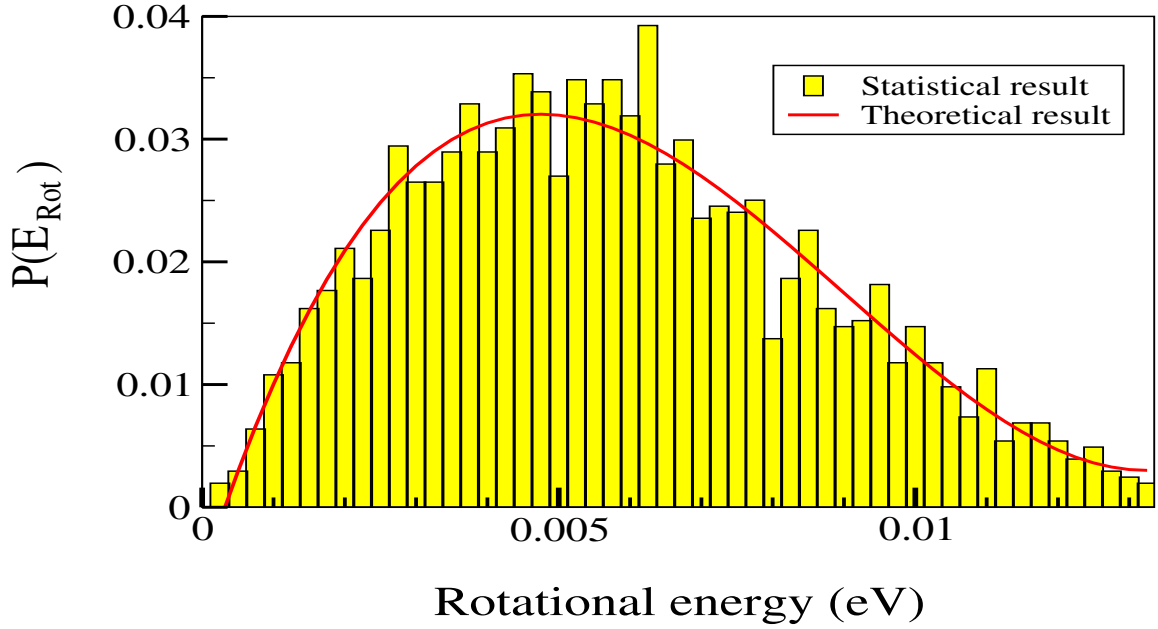


Fig 6.8: Distribution of rotational energy of the dimer

In Eqn (6.7) C represents the normalization constant and g_j represents the degeneracy.

The vibrational energy of the dimer corresponds to the zero point energy (ZPE) of the dimer.

$$ZPE = h\nu_0 / 2 = E_v \quad , \quad (6.8)$$

$$E_v = (1/2)\mu V_R^2 \quad , \quad (6.9)$$

where h is the Planck's constant, ν_0 is frequency, μ is the reduced mass of the dimer, and V_R is the relative vibrational velocity of the dimer. Eqn (6.9) assumes the initial vibrational phase of the dimer corresponds to the equilibrium position.

6.3. Predominant events in CVD dimer deposition

Many complex chemical reactions on a surface begin with simple elementary steps. These steps include adsorption on the surface, diffusion of the adsorbed atoms or molecules between binding sites, bond-breaking, insertion of

atoms or molecules and desorption of product molecules. In our present studies we have focused on the following events and their probabilities.

- Chemisorption
- Scattering/Reflection
- Desorption

6.3.1. Chemisorption

The initial conditions for the substrate and the carbon dimer are selected as discussed above. The position and the force on each atom in the system is determined by solving the Newtons equations of motion. The potential energy of the system is monitored after every integration step. A sudden drop in the potential energy of the system was noticed as the dimer approaches the substrate indicating the bond formation between the carbon dimer and the radical site. The trajectory calculations are carried out for an additional time of 1 *ps* after the dimer has reached the surface. Chemisorption of the carbon dimer is said to have occurred if the adsorbed atom undergoes ten or more inner turning points with respect to motion in the surface normal direction and the distance between the radical site and one of the carbon atoms of the dimer is within a cut-off radius of 2 Å of the radical site. The chemisorption probability is determined by running 50 trajectories keeping the input parameters, such as the incidence angle, rotational angle, translational energy, rotational energy and impact parameter constant and averaging over other factors such as the thermal vibrations of the lattice, vibrational phase angles of the lattice, rotational plane of the dimer and

initial orientation of the dimer. Figure 6.9 gives the variation of potential energy of the system V , the Z coordinate of the center of mass of the dimer, and the

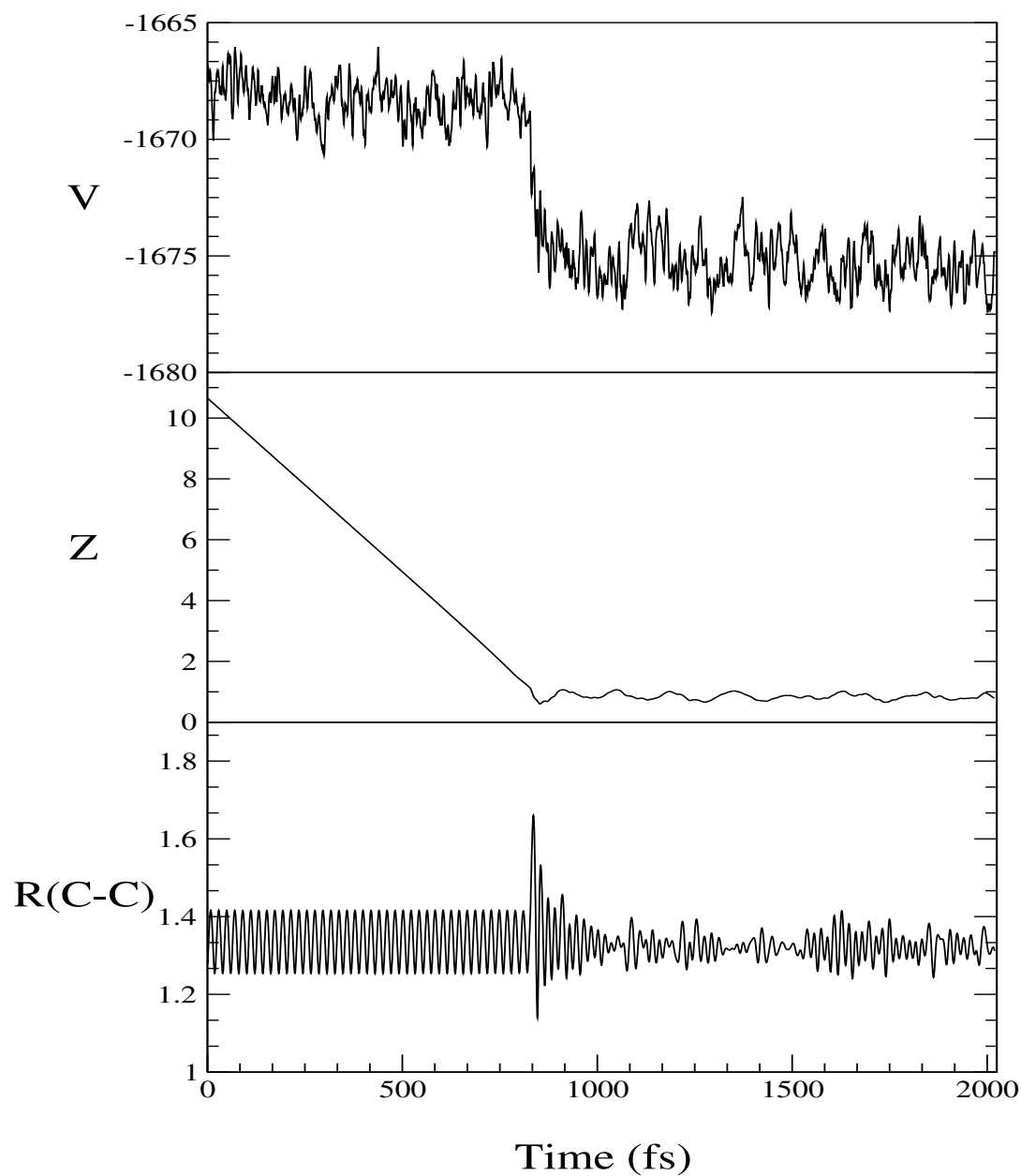


Fig 6.9: Variation of system's potential energy, Z coordinate of COM of the dimer distance between carbon atoms of the dimer (chemisorption event)

distance between two carbon atoms of the dimer, R , as a function of time. It can be seen from Figure 6.9, there is a sudden drop in the potential energy of the system by ~ 6 eV at the instance of bond formation.

6.3.2. Scattering

Scattering of the dimer is said to have occurred if the dimer executes only

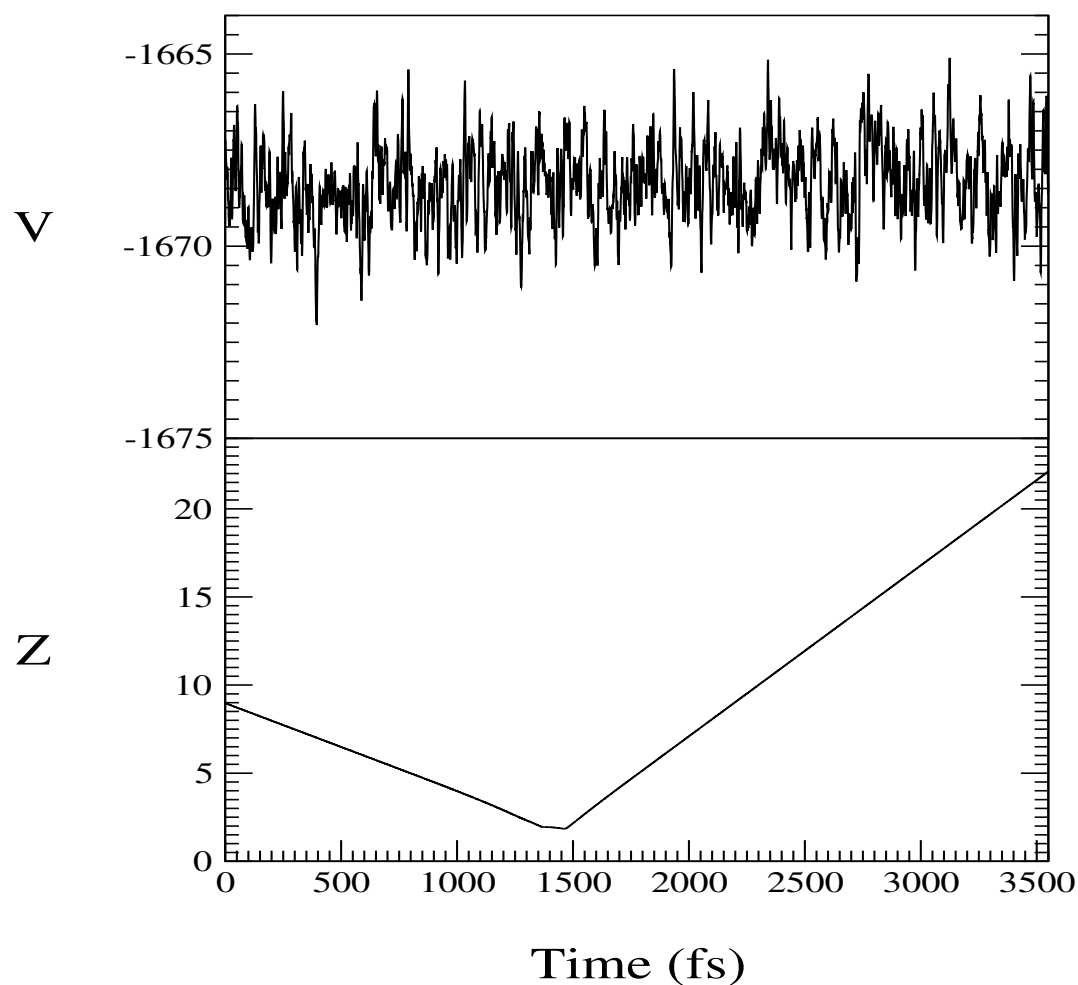


Fig 6.10: Variation of system's potential energy and Z coordinate of COM of the dimer (scattering event)

one inner turning point on the surface of the (100) lattice and then bounces back. The system energy is monitored over every integration time step and there is found to be no drop in the potential energy of the system as there is no bond formation between the incoming dimer atoms and the atoms of the diamond surface. The scattering probability is determined using the same procedure as used for chemisorption. Figure 6.10 gives the variation of potential energy of the system, V , and the Z coordinates of the center of mass of the dimer as a function of time. It can be seen from Figure 6.10, there is no change in the potential energy of the system as a result of scattering of the dimer.

6.3.3. Desorption

In the desorption event, the carbon dimer comes to the surface of the lattice, gets adsorbed without appreciable change in the potential energy and then desorbs back after a few oscillations. The probability of desorption is determined by running 50 trajectories keeping the incidence angle, rotation angle, translational velocity, rotational velocity and impact parameter constant and averaging over the thermal vibrations and vibrational phase angle of the lattice. Desorption is said to have occurred if the carbon dimer (C_2) executes less than 10 inner turning points on the diamond surface and bounces back. There is no change in the potential energy of the system.

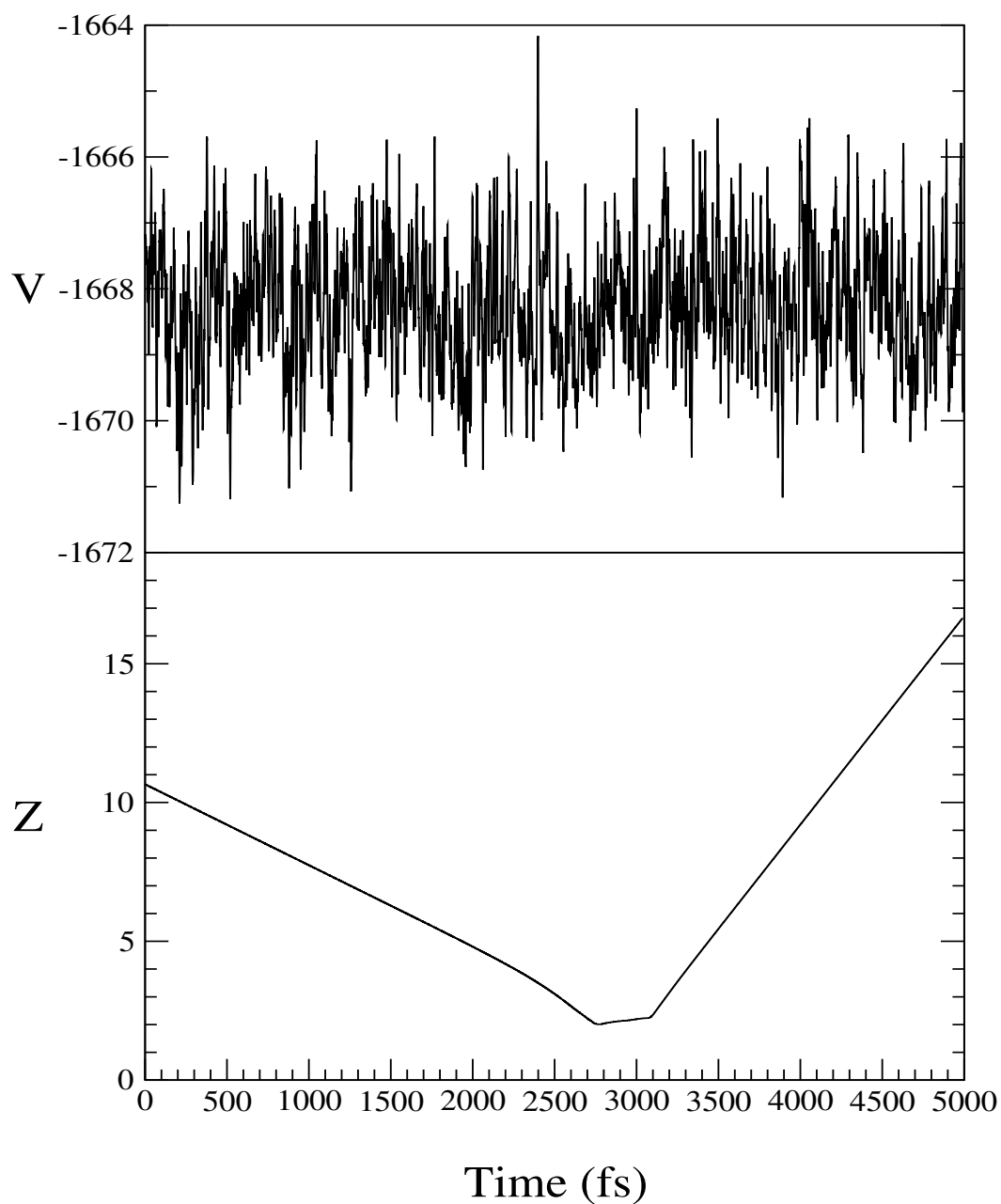


Fig 6.11: Variation of system's potential energy and Z coordinate of COM of the dimer (desorption event)

Figure 6.11 shows the potential energy of the system and Z coordinate of the dimer's center of mass with time for the desorption event. As can be seen from the plot, the dimer stays on the substrate for greater period of time as compared to that for a scattering event.

CHAPTER 7

NEURAL NETWORKS (NN) FOR EVENT PROBABILITY PREDICTION

The probabilities of various events, such as, chemisorption, scattering and desorption determined by MD simulation has been used to implement a neural network and subsequently use that neural network to predict the probabilities of these events as well as to find the effect of various input parameters on these probabilities. In studies described in Chapters 8, three separate networks have been implemented, one each, for predicting the probabilities of the three events. This chapter discusses the architecture of the network used, the structure of the input and output data set to the network, the total number of data sets used, and the procedure for implementation of the network, namely, training and testing of the network.

7.1. Architecture and working of the neural network

The neural network used in this investigation for predicting the probabilities of various events that occur during carbon dimmer (C_2) deposition on a diamond (100) surface is a multilayered feed forward network. The network has two layers, the first layer is a hidden layer that has 50 hidden neurons and a tansigmoid transfer function associated with every hidden neuron in the layer. The second layer is the output layer that has one neuron and a pure-linear transfer function

associated with it. The output from this second layer is the probability of a particular event and this forms the final output of the neural network.

The input vector to the neural network has five components, namely, the incidence azimuthal angle of the dimer (θ), the rotation angle (Φ), impact parameter (b), translational energy of the dimer (E_{Trans}), and rotational energy of the dimer (E_{Rot}). The output vector of the neural network has a single component which forms the probability of a particular event predicted by the neural network.

7.2. Implementation of the neural network

The implementation of the neural network for predicting the probabilities is carried over in two stages, namely, the training stage and the testing stage. The network is trained using Levenberg-Marquardt algorithm that employs a procedure known as early stopping [69]. In this procedure, the training of the neural network was met within approximately 20 iterations or epochs. The total input data containing 2000 data sets to the network is divided into two subsets, of which 85% of the data (1700 data sets) becomes the training set, which was used for training the network, and the remaining 15 % (300 data sets) of the data becomes the validation set. Fifty different neural networks were trained by random selection of the 85% of the training data and the average of outputs of these 50 networks is computed to arrive at the predicted probabilities.

It may be noted that during initial stage of each training, the error on the training and validation sets decrease. But, when the network starts overfitting, the error on the training set continues to decrease while that on the validation set starts increasing. When the error on the validation set begins to increase for a

specified number of iterations, it indicates the network is attempting to overfit and so the training is stopped. Such an early stopping procedure has been successfully used to prevent the network from overfitting [69].

After the network has been trained successfully, it is tested by supplying it with a set of input data to predict the output. If the network has been trained properly, it will be able to predict an output that matches closely with the desired output.

CHAPTER 8

PREDICTION OF EVENT PROBABILITIES – NEURAL NETWORK VS MD SIMULATION

In this chapter we report results of MD simulations for predicting the probabilities of various events, such as, chemisorption, scattering, and desorption that occur during the deposition of carbon dimer (C_2) species on a diamond (100) surface in the CVD process. We shall use these results to train the neural network and determine the underlying relationships between the five input parameters of the dimer and each of the three event probabilities.

8.1. Data points generation for neural networks

The five input parameters used in the synthesis of diamond by CVD process, namely, the incidence angle (θ), rotation angle (Φ), impact parameter (b), translational energy (E_{Trans}), and rotational energy (E_{Rot}) of the dimer forms the input vector for the neural network and the corresponding event probabilities forms the output vector of the neural network. A total of 2000 data points are used for training and testing the neural network. Every point for the neural network is generated by running 50 MD trajectories. All the five input parameters were kept constant during these 50 trajectories. The probability of occurrence of each of the three events were estimated at the end of these 50 trajectories by

taking the ratio of the number of times a particular event has occurred to the total number of trajectories computed.

8.2. Training and testing of the neural network

As mentioned in Chapter 7, the implementation of the neural network involves training and testing. First, the five input parameters and the output probabilities were normalized to make the range to lie between -1 and +1. Normalizing is done using the formula

$$p_n = \frac{2(p - p_{\min})}{(p_{\max} - p_{\min})} - 1, \quad (8.1)$$

where p is the variable to be scaled, p_{\min} and p_{\max} are the minimum and maximum values of each variable in the input or output vectors for the entire database consisting of all the points. p_n is the normalized value corresponding to p . 85 % of the normalized data have been used for training and the remaining 15% is used for the validation of the network. 50 neural networks were generated by a random selection of 85% of the training data, and the average of the outputs of these 50 networks is computed to obtain the final predicted probabilities. The training of the neural network was accomplished within approximately 20 iterations or epochs. The initial weight matrices for each training were randomly chosen. This is done to enable the network to get trained for any randomness. Each neural network was trained using supervised learning mentioned in Chapter 3. Early stopping was used to prevent the network from overfitting [69]. After each neural network has been trained, the network was tested with a test data set to see whether the network is able to predict the outputs correctly.

Figures 8.1 through 8.3 show the training and testing plots for the three events, namely, chemisorption, scattering, and desorption for one of the neural networks. The scatter present in the training and testing plots is because of the uncertainty occurring due to averaging over just 50 trajectories for calculating the probabilities of the events.

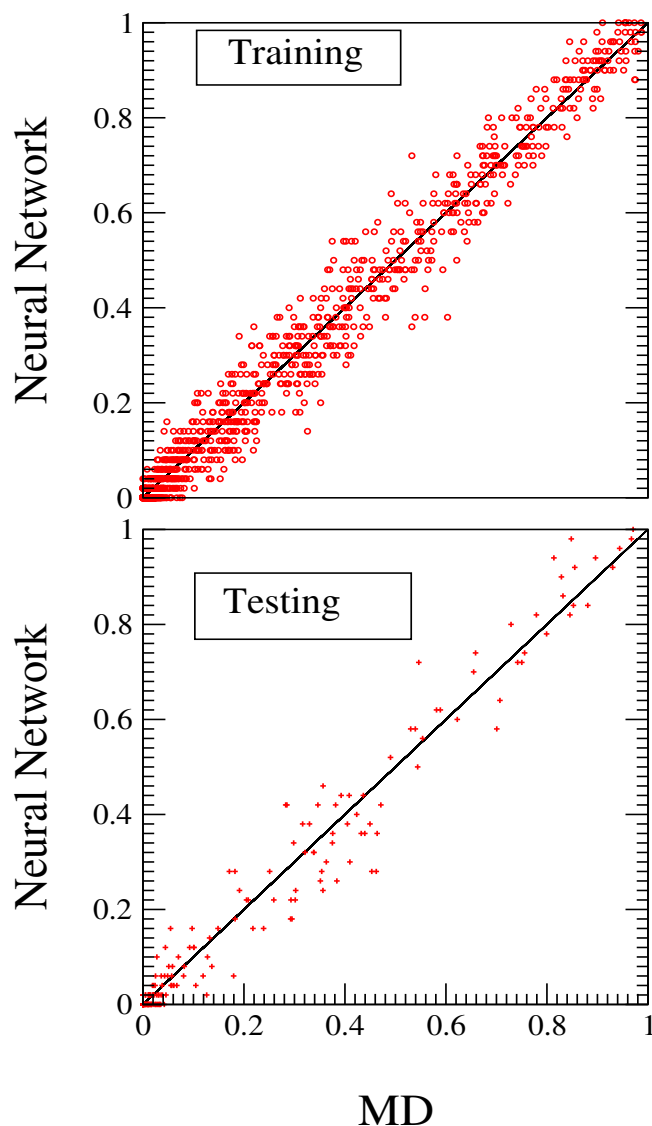


Fig 8.1: Neural network training and testing plots for the probability of chemisorption for one neural network

The scatter in the plots can be greatly minimized by averaging over a larger number of MD trajectories, say 500 trajectories per data point instead of 50 trajectories per data point. For example, if we average over 500 MD trajectories, and assume one chemisorption event occurred then the statistical uncertainty involved here is calculated using Eqn (8.2) to be 0.001999. Now, let us take the

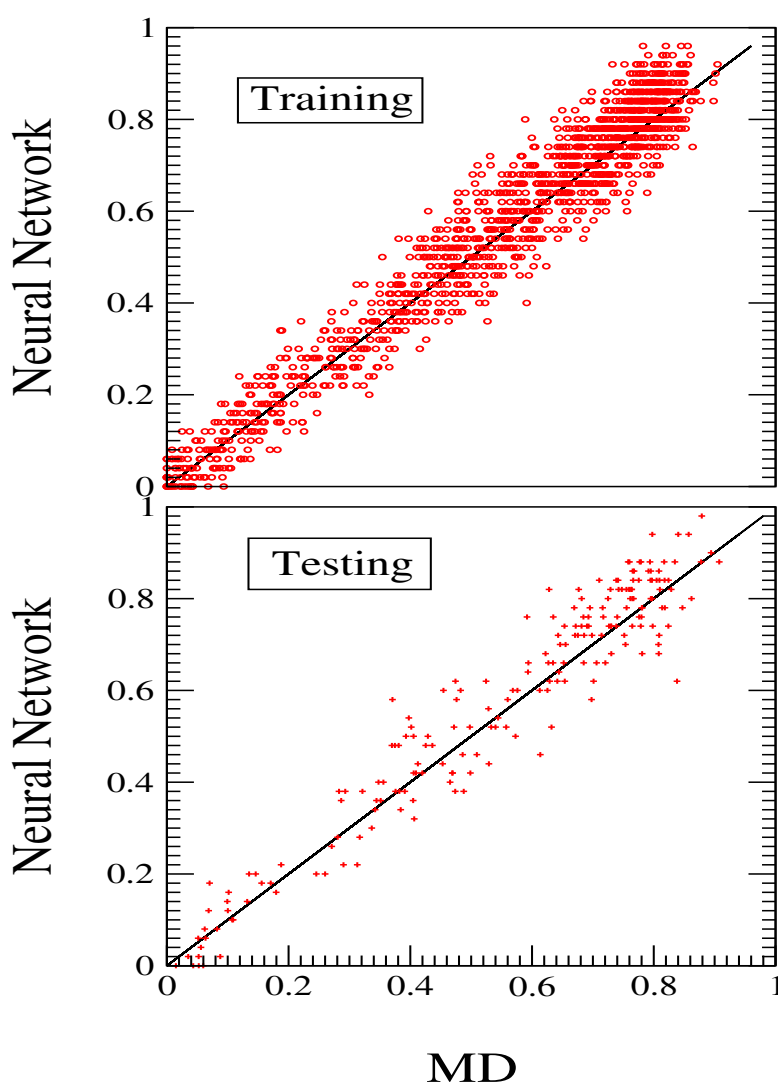


Fig 8.2: Neural network training and testing plots for the probability of scattering for one neural network

present case in which we average over 50 trajectories, and one chemisorption event occurred, the statistical uncertainty in this case is 0.0197. We see that the statistical uncertainty reduces by ten times if the number of trajectories is increased.

As mentioned in Chapter 3, three individual neural networks have been used for predicting the probabilities of the three events, one network for each event probability.

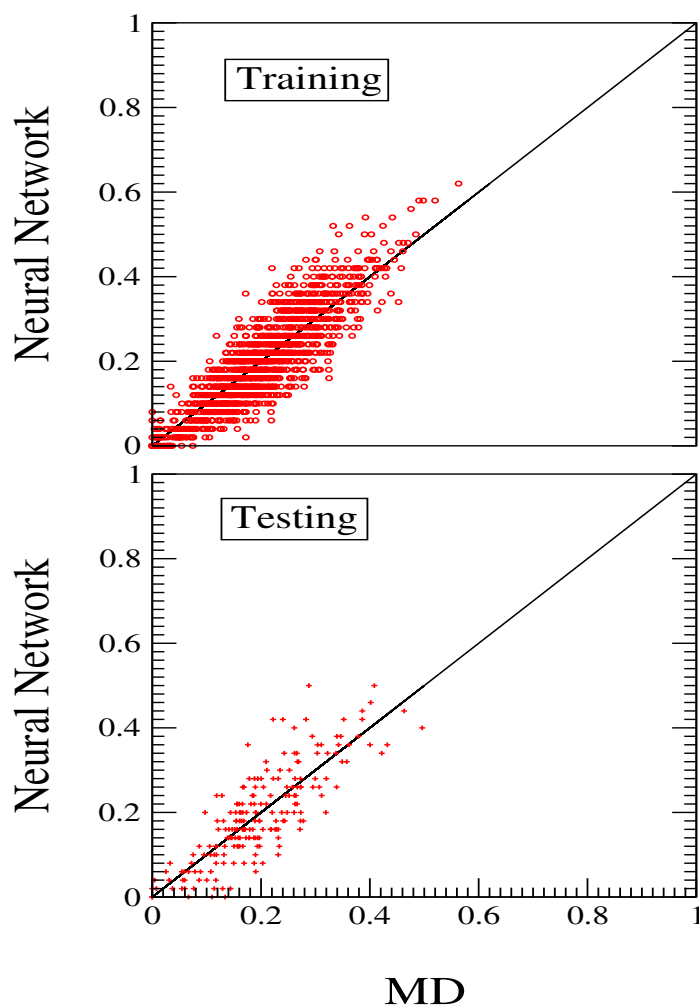


Fig 8.3: Neural network training and testing plots for the probability of desorption for one neural network

The rms error in the training was found to be 0.0422 for the chemisorption probability network, 0.0561 for the scattering probability network and 0.0550 for the desorption probability network. The rms error during testing was found to be 0.0512 for the chemisorption probability network, 0.0695 for the scattering probability, network and 0.0630 for the desorption probability network.

8.3. Effect of input parameters on event probabilities: Neural network Versus MD predictions

The effect of the five input parameters, on the probabilities of chemisorption, scattering, and desorption has been studied using MD simulations. Subsequently, neural networks were used to predict the relationship existing between the input parameters and the event probabilities. In this section we will investigate the predictions made by the neural networks and compare their predictions with MD simulation results.

8.3.1. Effect of incidence angle (θ)

The effect of incidence angle (θ) on the three probabilities was studied using MD simulations by running trajectories in which the other four input parameters are maintained constant. For every value of the incidence angle of the dimer 50 trajectories were run in order to average over the thermal vibrations and vibrational phase angles of the lattice. The probabilities of the three events were determined as described in Section 8.1. The input parameters for which MD trajectories were run are as follows: $\Phi = 110^\circ$, $b = 1 \text{ \AA}$, $E_{\text{Trans}} = 0.124 \text{ eV}$, $E_{\text{Rot}} = 0.052 \text{ eV}$ and the incidence angle is varied from $\theta = 0^\circ$ to the maximum incidence angle $\theta_{\text{max}} = 23^\circ$ in steps of 2° . The neural networks that were trained (as described in Section 8.2)

are used to simulate the event probabilities for the same input data and the results are plotted in Figure 8.4 along with the MD results. It can be seen from Figure 8.4,

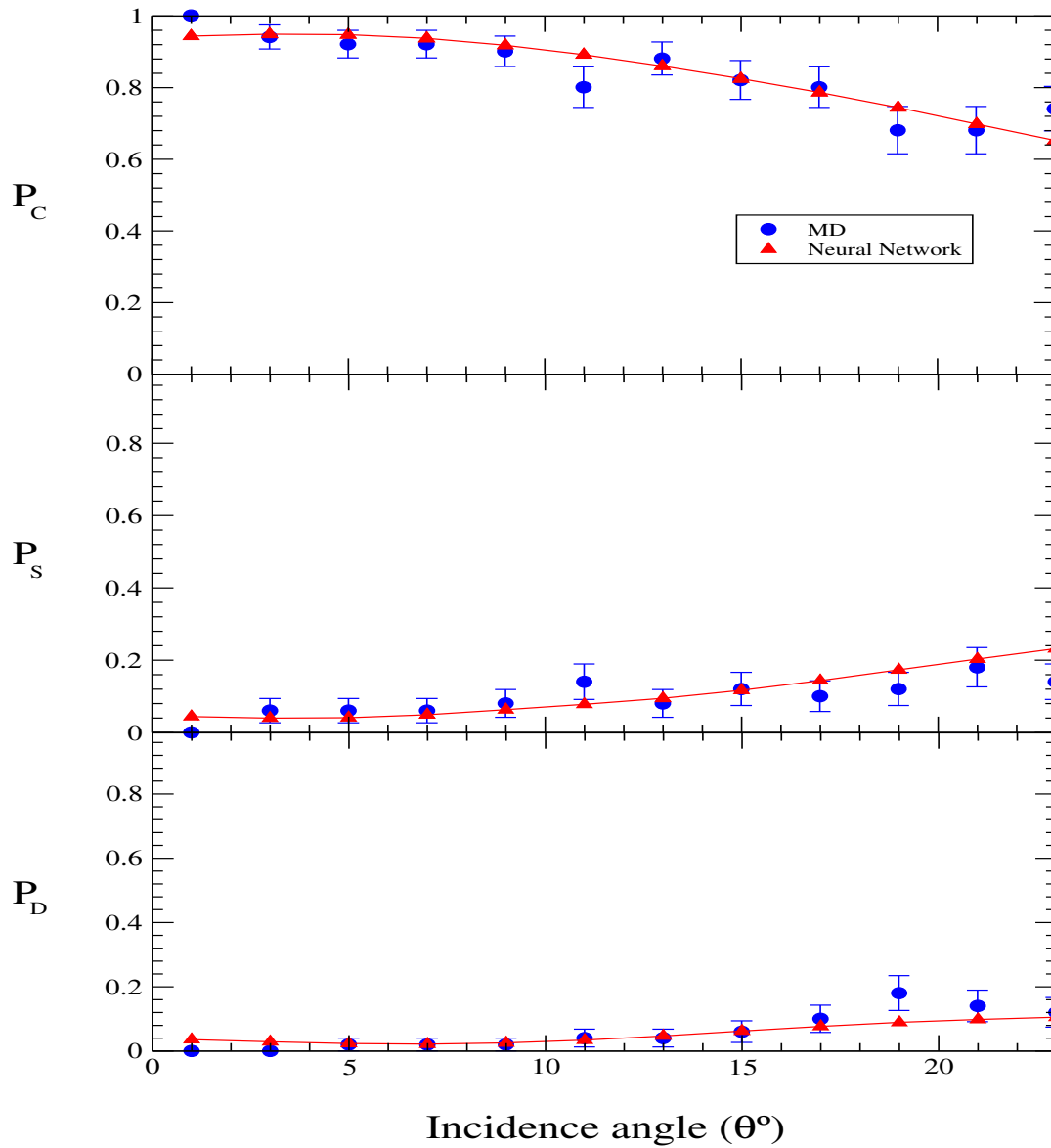


Fig 8.4: Effect of incidence angle on chemisorption, scattering and desorption probabilities - MD and Neural network predictions. The error bars represent one sigma limit of statistical uncertainty in the MD results.

the results predicted by MD calculations and neural network agree well with each other. The chemisorption probability, scattering probability, and desorption probability are represented as P_C , P_S , and P_D , respectively, in the graphs. The

statistical error present in MD is found to be one sigma limit of uncertainty. It is calculated using the formula [36]

$$\nabla = [(N - N_R) / NN_R]^{1/2} * \sigma(P). \quad (8.2)$$

Here, N represents the total number of trajectories, N_R represents the number of times a particular event has occurred, and $\sigma(P)$ represents the event probability. For example, say that out of the 50 trajectories ran, 40 events are chemisorption, then the one sigma limit of uncertainty in MD using Eqn. (8.2) is 0.0565.

8.3.2. Effect of rotation angle (Φ)

The effect of rotation angle on the event probabilities is studied using MD simulations using the same procedure. In this case, all parameters except the rotation angle (Φ) are kept constant for all trajectories. The rotation angle is varied from 10° to 360° in steps of 20° for every 50 trajectories. The input parameters for which MD trajectories were run are given as: $\theta = 11^\circ$, $b = 1 \text{ \AA}$, $E_{\text{Trans}} = 0.06 \text{ eV}$, $E_{\text{Rot}} = 0.052 \text{ eV}$. The same input data sets that were used for running the MD trajectories are used as the test data set for the neural networks and the average neural network output is plotted along with the MD results in Figure 8.5. The error bars in the figure corresponds to the statistical error in MD and is computed using

Eqn (8.2). We notice that the MD results and the neural network results agree well with each other.

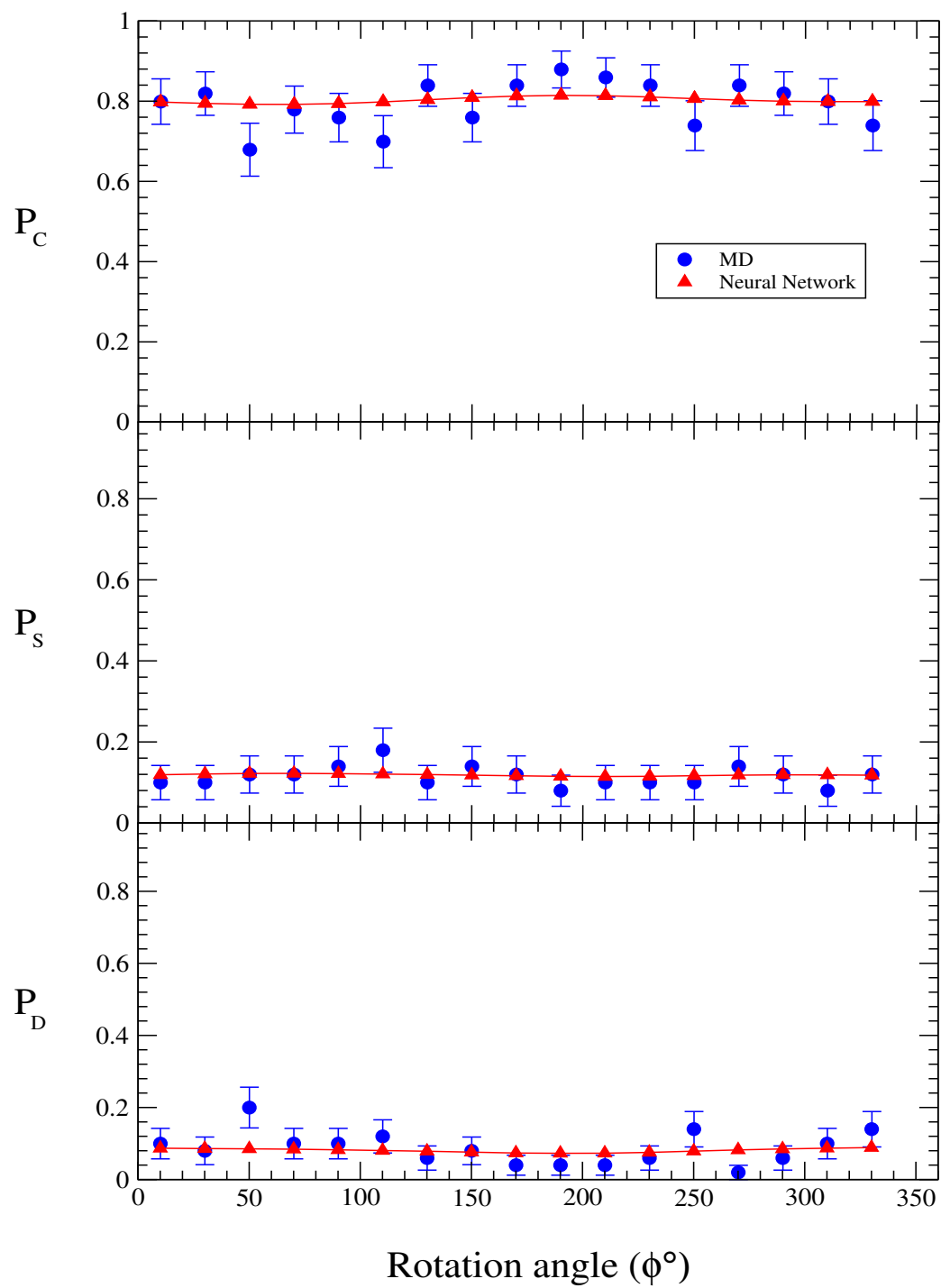


FIG 8.5: Effect of rotation angle on chemisorption, scattering and desorption-probability MD and Neural network predictions. The error bars represent one sigma limit of statistical uncertainty in the MD results.

8.3.3. Effect of impact parameter (b)

The effect of impact parameter on the three event probabilities are determined using MD simulations using the same procedure as described in Section 8.3.1.

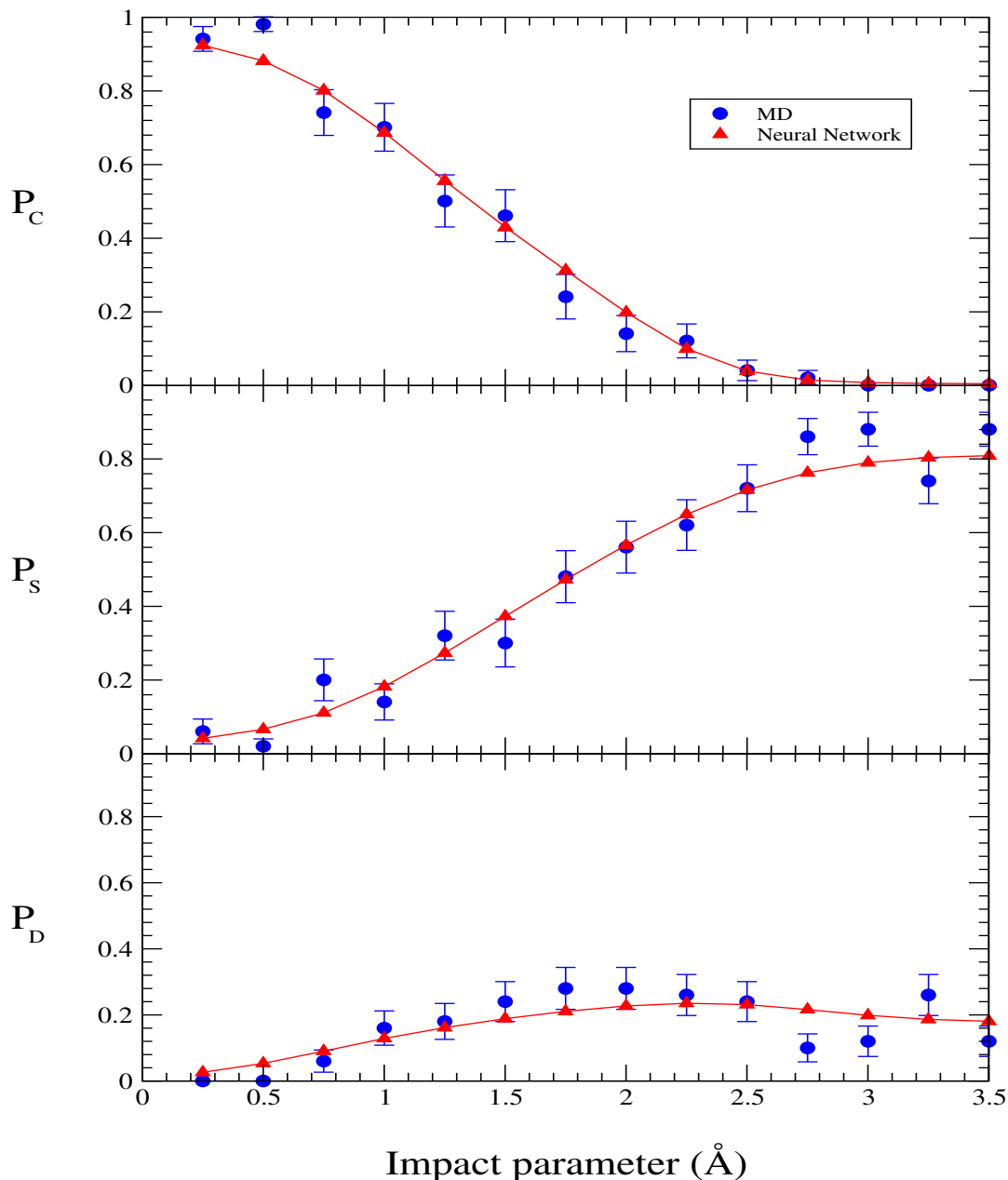


FIG 8.6: Effect of impact parameter on chemisorption, scattering and desorption probabilities MD and Neural network predictions. The error bars represent one sigma limit of statistical uncertainty in the MD results.

The impact parameter is varied from 0.25 Å to maximum impact parameter $b_{\max} = 3.5$ Å in steps of 0.25 Å for every 50 MD trajectories. The values of other input parameters are as follows: $\theta = 17^\circ$, $\Phi = 310^\circ$, $E_{\text{Trans}} = 0.06$ eV and $E_{\text{Rot}} = 0.052$ eV. The neural networks are now used for predicting the event probabilities by using the same test data as used for running the MD calculations. The results are plotted along with MD results and uncertainty associated with MD in Figure 8.6. It can be seen that the output of the neural network is in accordance with that of MD.

8.3.4. Effect of translational energy of the dimer (E_{Trans})

The effect of translational energy of the dimer on each of the three event probabilities were determined using MD using the same procedure as described above for the other parameters. The values of input parameters are as follows: $\theta = 11^\circ$, $\Phi = 110^\circ$, $b = 1$ Å and $E_{\text{Rot}} = 0.052$ eV. The same input data set is used for the neural networks. The average output of the networks and MD results along with statistical error associated with MD are shown in Figure 8.7. Here again, we note a good agreement between MD and NN.

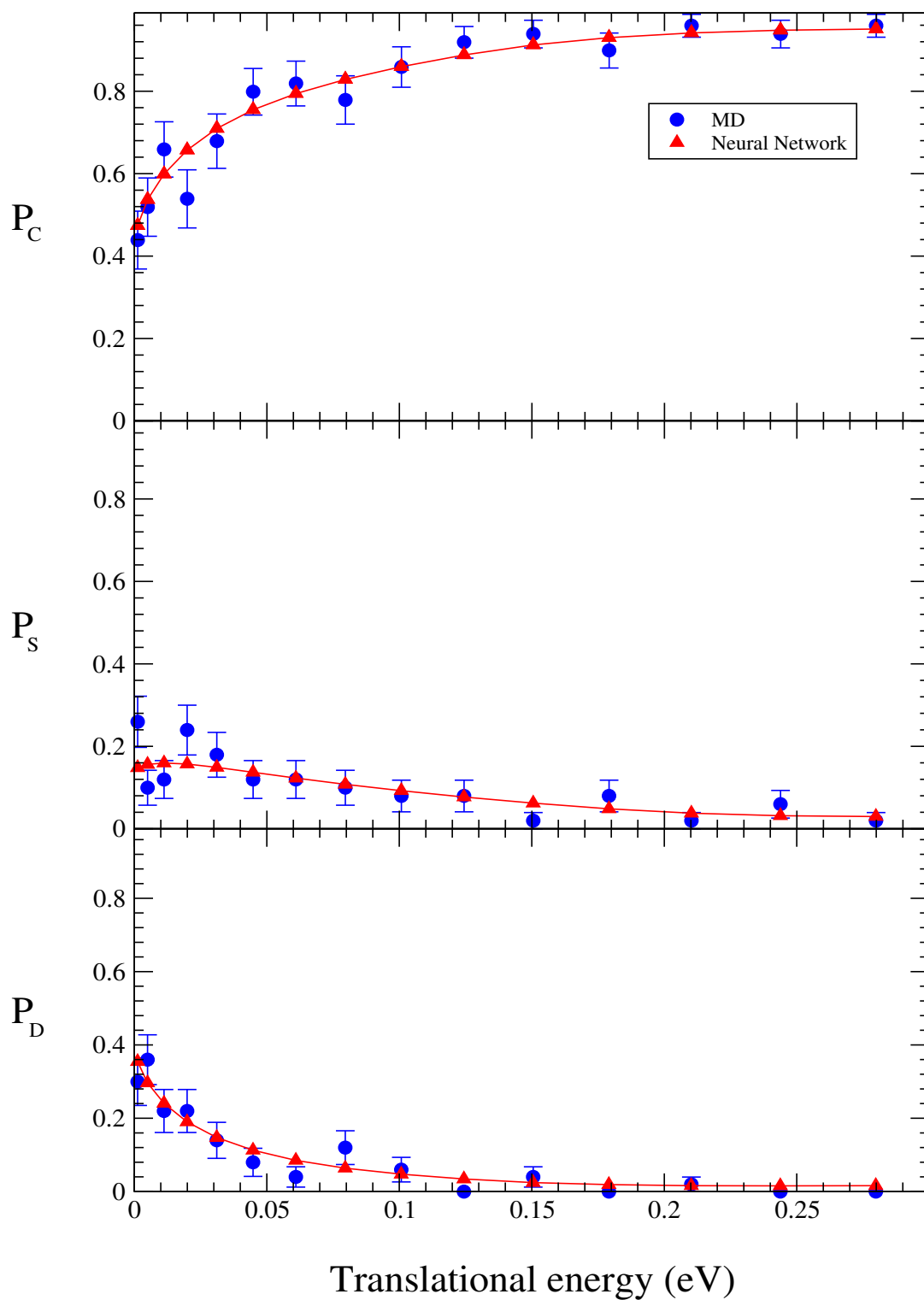


Fig 8.7: Effect of translational energy on chemisorption, scattering and desorption probabilities – MD and Neural network predictions. The error bars represent one sigma limit of statistical uncertainty in the MD results.

8.3.5. Effect of rotational energy of the dimer (E_{Rot})

The effect of rotational energy of the dimer on three probabilities is determined using MD simulations and also using neural networks using the same

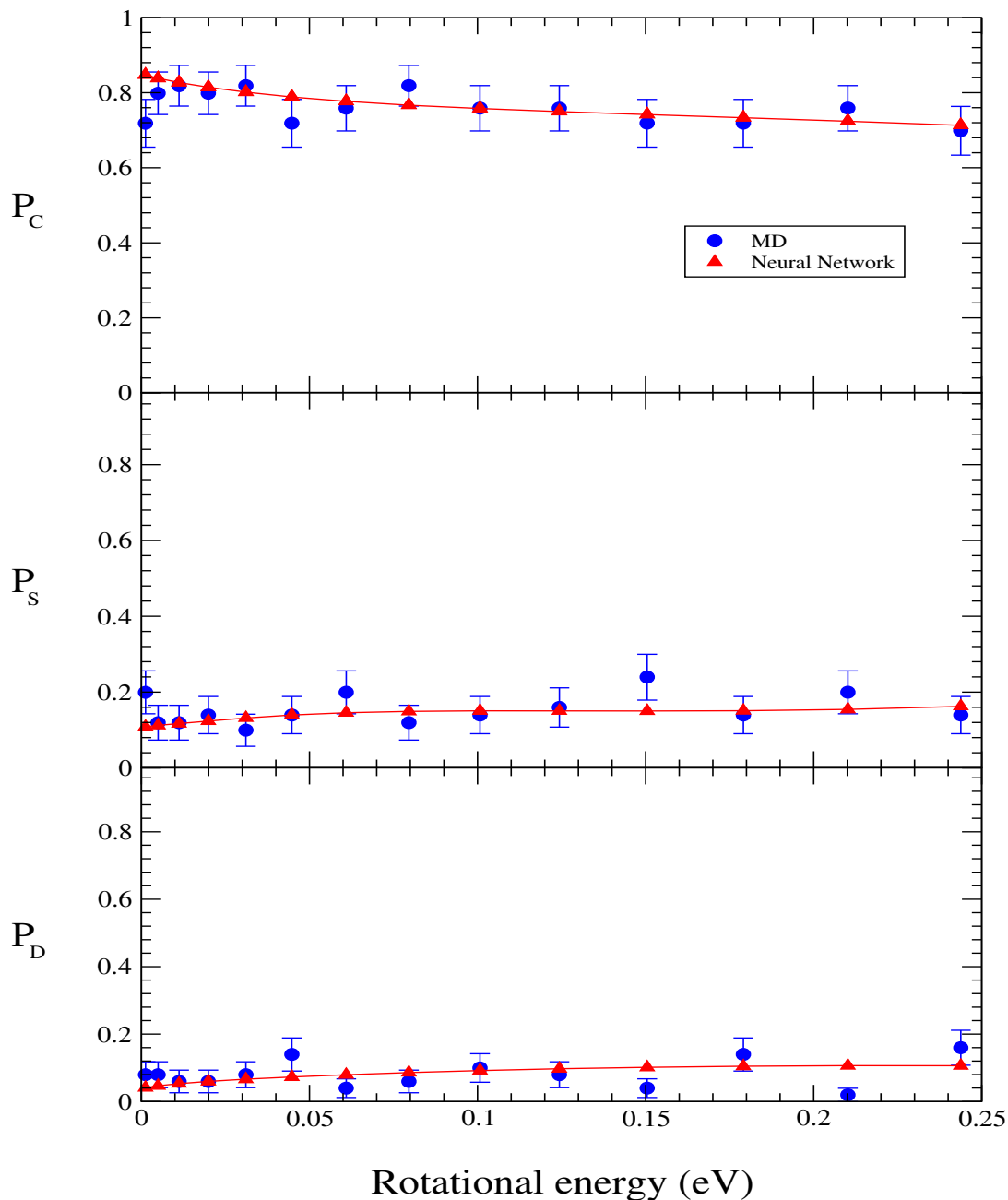


Fig 8.8: Effect of rotational energy on chemisorption, scattering and desorption probabilities – MD and Neural network predictions. The error bars represent one sigma limit of statistical uncertainty in the MD results.

procedure as described in section 8.3.1. The input data set for MD calculations are as follows: $\theta = 17^\circ$, $\Phi = 110^\circ$, $b = 1 \text{ \AA}$, $E_{\text{Trans}} = 0.06 \text{ eV}$. The neural networks are tested to predict the output for the same data set and the results of MD and neural network are shown in Figure 8.8. We note that the neural network and MD results agree well with each other.

8.4. Statistical uncertainty: Neural network Vs MD

The results given by molecular dynamics simulation have a statistical uncertainty that can be calculated using Eqn. (8.2). Figures 8.4 through 8.8 show the MD results with error bars to indicate the one sigma limit of statistical uncertainty in the MD calculations. The neural network plots are obtained by averaging over 50 sets of neural network matrices. Therefore, the neural network predictions also have statistical errors associated with them, but, the error in neural network prediction is very small compared to MD (See Figure 8.9). The figure shows the neural network predictions and MD predictions along with the error bars to show the one sigma limit of statistical uncertainty associated with each case. The error bars in dotted lines represent the statistical noise associated with neural network, and the error bars in solid lines represent the statistical error associated with MD. We infer from Figure 8.9 that the functional relationship between various input parameters and different event probabilities predicted by the neural network are continuous and have less statistical uncertainty associated with them than do the MD results.

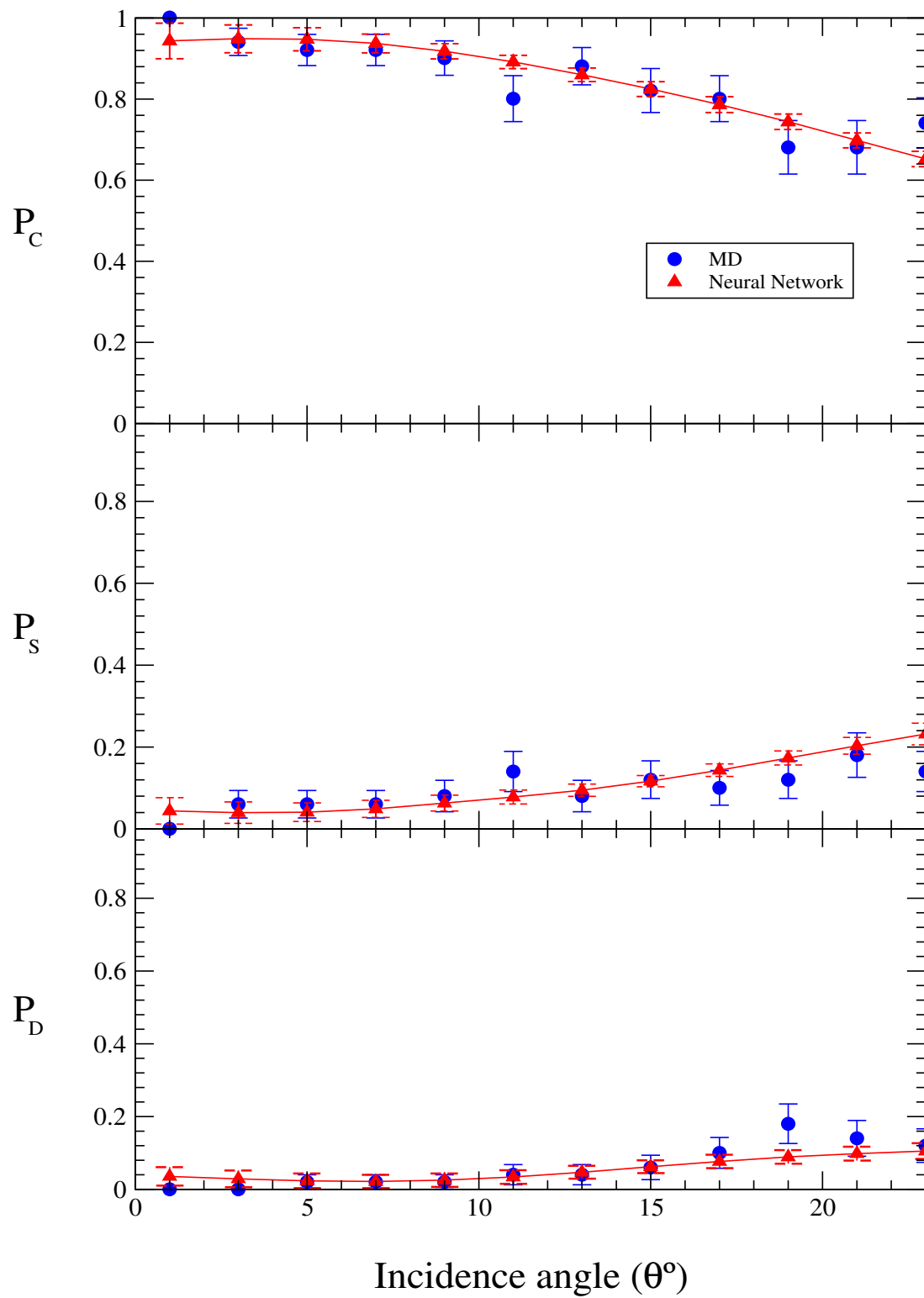


Figure 8.9: Comparison of statistical uncertainty in neural network and MD

8.5. More results from neural networks

In Section 8.3, we have seen that neural network is able to predict the underlying relationship existing between the incidence angle (θ), rotation angle (Φ), impact parameter (b), translational energy (E_{Trans}) and rotational energy (E_{Rot}) of the dimer, and the probabilities for chemisorption, scattering and desorption. After training the neural network, it is easy to compute the probabilities of different events for arbitrary sets of input parameters. In Figures 8.10 through 8.14, we present additional results given by the trained neural network. The time taken by the neural network for predicting the relationship between each of the input parameters and the three event probabilities is approximately 3 minutes, whereas MD simulation takes 550 minutes (~9.2 hours) to study the influence of a single parameter on three event probabilities. So, it is easy to note that the computation of the Figures 8.10 through 8.14 by MD simulations would require hundreds of CPU hours in contrast to a few minutes by the trained neural networks.

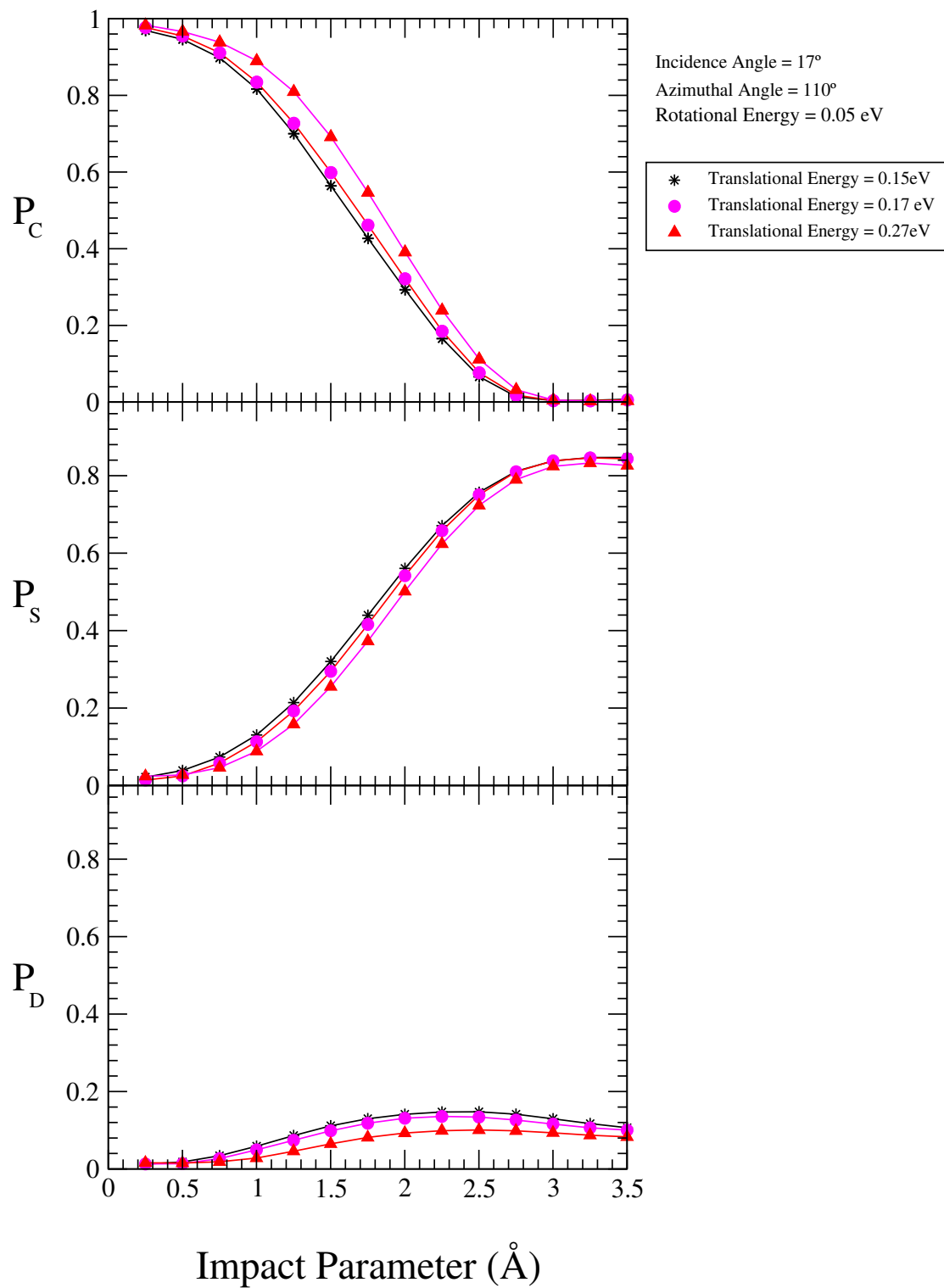


Fig 8.10: Effect of impact parameter on event probabilities for various translational energies of the dimer- NN predictions

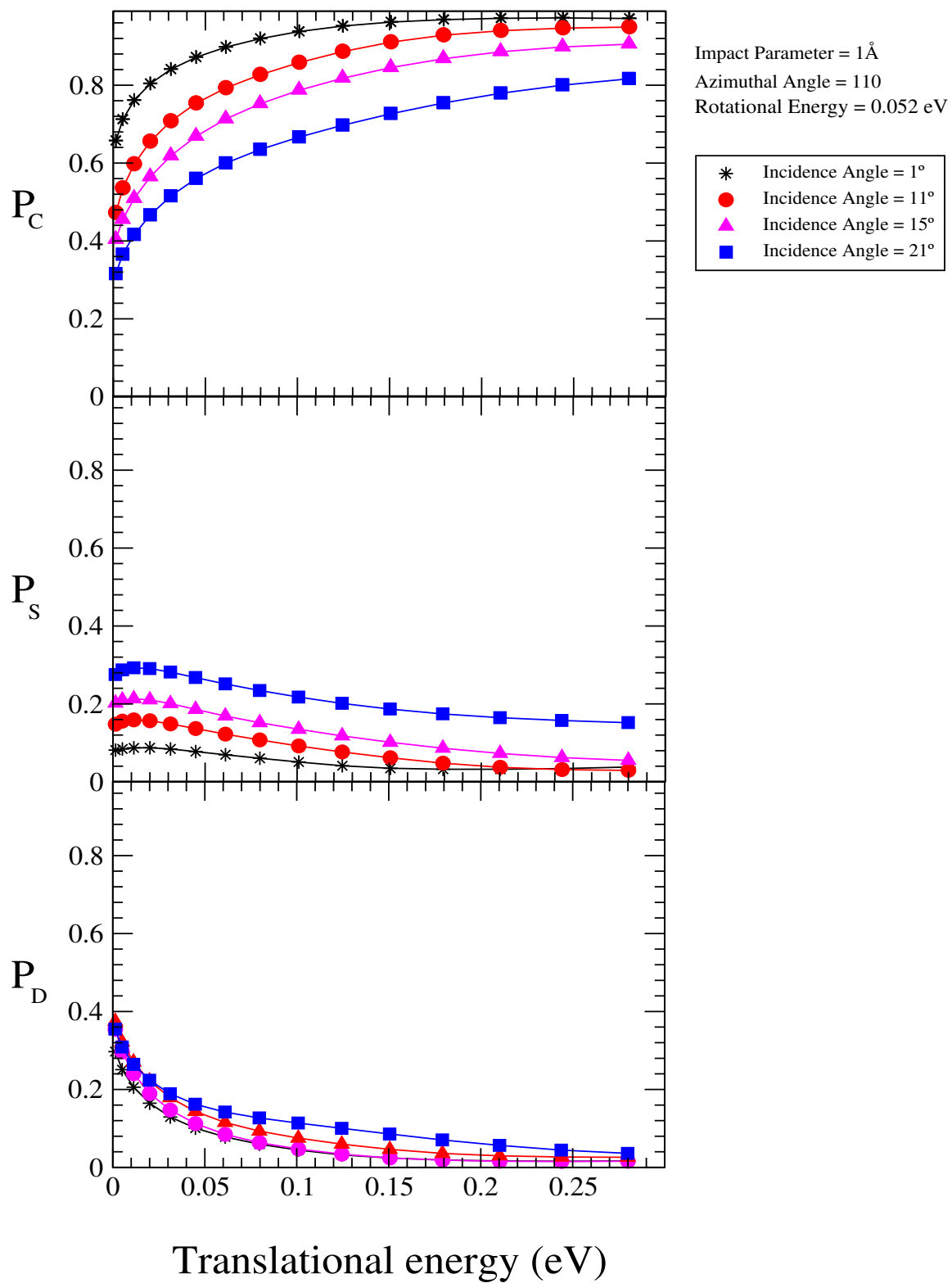


Fig 8.11: Effect of translational energy on event probabilities for various incidence angles of the dimer- NN predictions

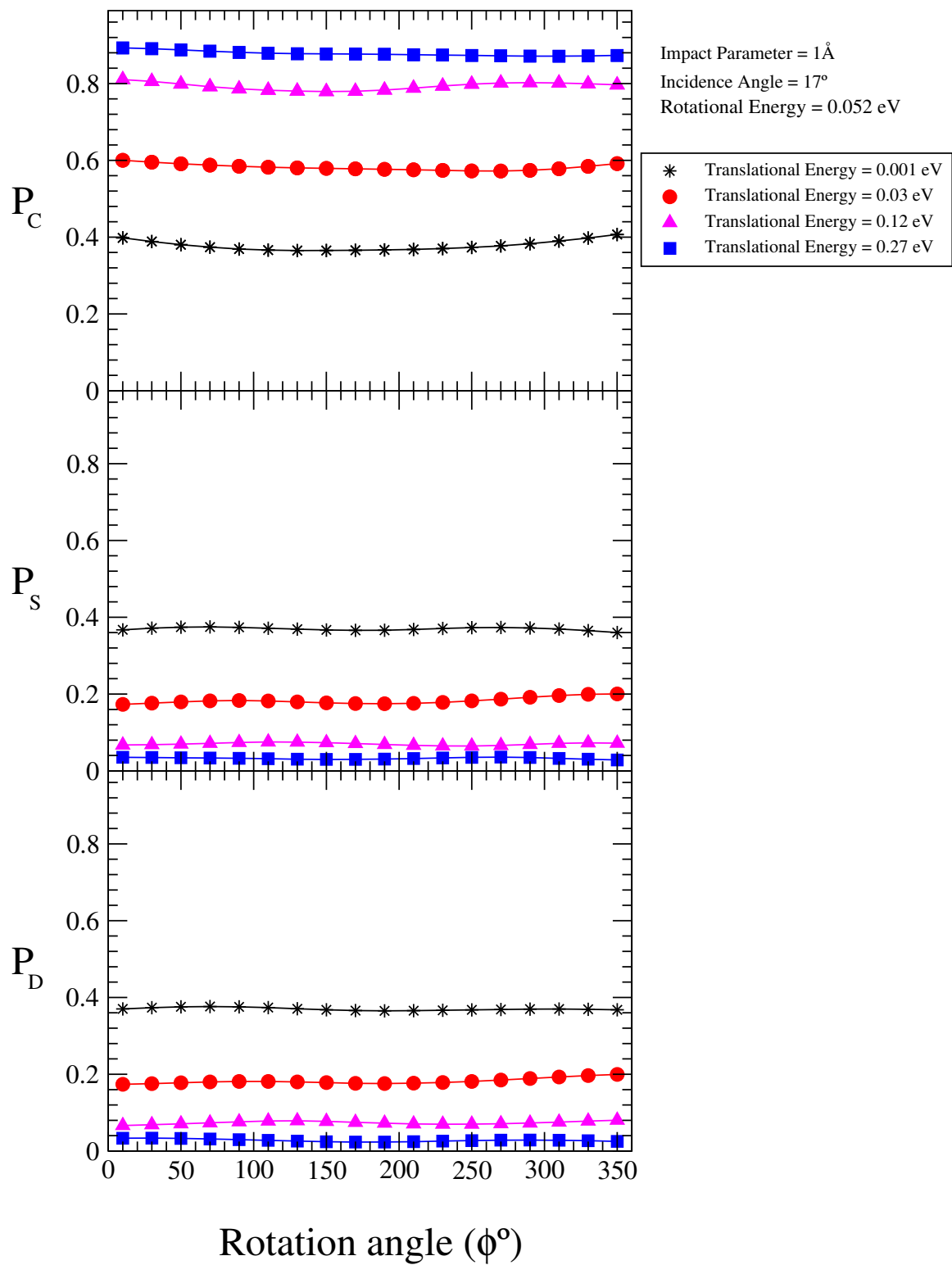


Fig 8.12: Effect of rotation angle on event probabilities for various translational energies of the dimer – NN predictions

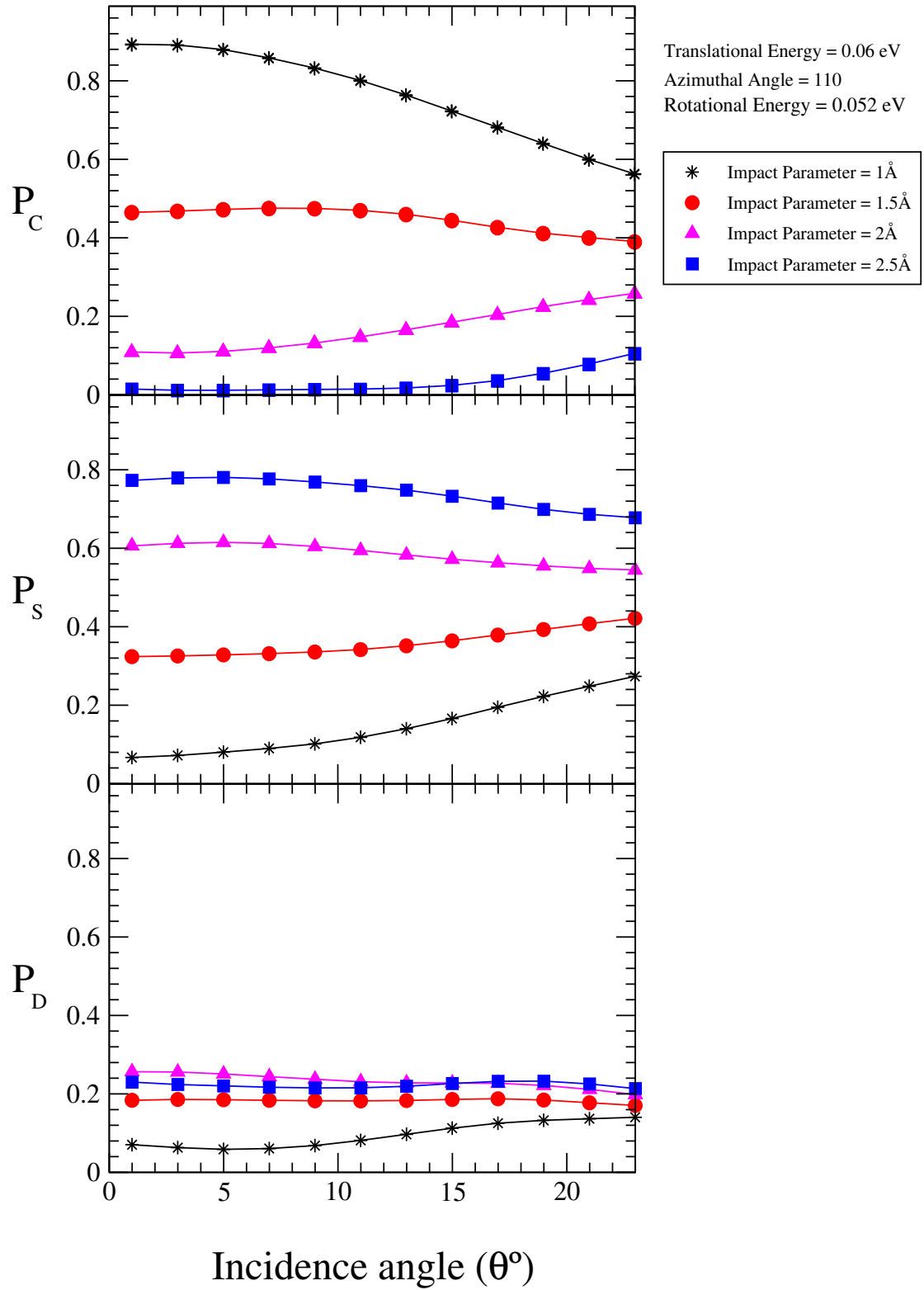


Fig 8.13: Effect of incidence angle on event probabilities for various impact p parameters – NN predictions

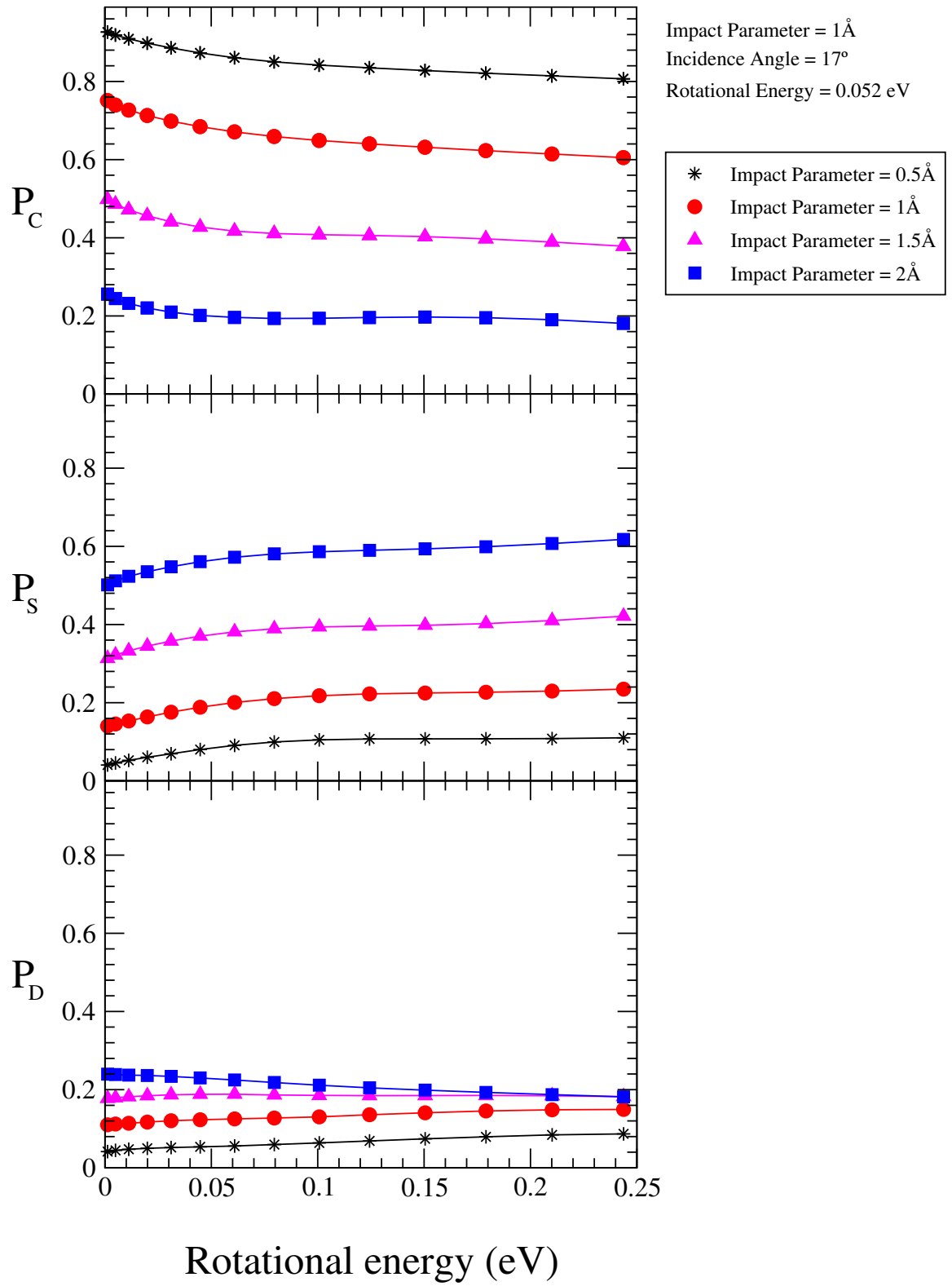


Fig 8.14: Effect of rotational energy on event probabilities for various impact parameters – NN predictions

CHAPTER 9

CONCLUSIONS AND FUTURE INVESTIGATION

MD simulations were conducted to generate the initial data required for training the neural network. The neural networks have been successfully applied to study the effects of five input parameters (incidence angle (θ), rotation angle (Φ), impact parameter (b), translational energy (E_{Trans}) and rotational energy (E_{Rot}) of the dimer), on the probabilities of three events (chemisorption, scattering, and desorption events), that occur during the deposition of carbon dimer (C_2) onto the diamond (100) surface in a CVD process for thin film growth. The conclusions of this study and future investigations are presented in the following.

9.1. Conclusions

1. Neural networks (NN) can be used effectively to predict the underlying relationships between the five input parameters of the dimer, and the probabilities of three events outlined above.
2. The chemisorption probability is found to decrease with increase in impact parameter (b). The scattering probability and desorption probability are found to increase with the impact parameter (b).
3. The chemisorption probability is found to increase with increase in the translational energy (E_{Trans}) of the C_2 dimer but the scattering and desorption

probabilities are found to decrease with the increase in the translational energy (E_{Trans}) of the dimer.

4. The chemisorption probability, scattering probability, and desorption probability are found to be independent of the rotation angle (Φ).

5. The chemisorption probability is found to decrease with increase in the incidence angle (θ) of the dimer. The scattering probability and desorption probability are found to increase with the incidence angle (θ) of the dimer.

6. The chemisorption probability is found to decrease with increase in the rotational energy (E_{Rot}) of the C_2 dimer but the scattering and desorption probability are found to increase with increase in the rotational energy (E_{Rot}) of the dimer.

9.2. Future Work

1. The approach presented in this investigation can be extended to investigate different types of reaction channels and mechanisms that occur during diamond film growth.

2. The neural network concept applied here can be extended to investigate the event probabilities of other types of growth species, such as CH_3 , C_2H_2 , and C_2H_4 .

3. With a slight modification to the neural network used in this study, the effects of the type of substrate used, lattice plane(s), temperature and pressure effects on event probabilities, reaction channels, and growth rates can be investigated to determine the optimum temperature and pressure conditions, and appropriate crystal planes for achieving high growth rates and better quality of the films deposited.

4. By training the network with more data sets for a particular event, say for example, insertion or hydrogen abstraction, the network can be strengthened to predict the probabilities of such rarely occurring events with high accuracy and less time.

5. The neural network approach used in this study can also be successfully applied to investigate the reaction channels leading to the growth of other thin films, such as polycrystalline silicon and gallium arsenide.

REFERENCES

1. Bundy, F.P, Hall H.T., Strong, H.M. and R.H. Wentorf, Jun. "Man – made diamonds," *Nature* **176** (1955). 1-7.
2. Mitura, S., "Nucleation of diamond powder particles in an RF methane plasma," *J. Cryst. Growth*. **80** (1987) 417-424.
3. Howard, W., Huang, D., Yuan, Frenkiach, M., Spear, K.E., Koba, R. and A.W. Phelps, "Syntehsis of diamond powder in acetylene oxygen plasma," *J. Appl. Phys.* **68** (1990) 1247-1251.
4. Frenkiach, M., Howard, W., Huang, D., Yuan, J., Spear, K.E., and R. Koba, "Induced nucleation of diamond powder," *Appl. Phys. Lett.* **59** (1991) 546-548.
5. Buerki, P. R. and Samuel Leutwyler, "Homogeneous nucleation of diamond powder by CO₂ –laser-driven gas-phase reactions," *J. Appl. Phys.* **69** (1991) 3739-3744.
6. Choy, K.L., "Chemical vapor deposition of coatings," *Progress in Materials Science* **48** (2003) 57-170
7. Celli, F.G., "Diamond chemical vapor deposition," *Annu. Rev. Phys. Chem.* **42** (1991) 643-684.

8. Yang, S.W., Xie, X., Wu, P. and K.P. Loh, "Chemisorption of C₂ biradical and acetylene on reconstructed diamond (111)-(2 x 1)," J.Phys. Chem. B. **107** (2003) 985-993.
9. Hukka, T.I. and T.A. Pakkanen, "Chemisorption of hydrogen on the diamond (100)2 X 1 surface: An *ab Initio* study," J. Phys. Chem. **98** (1994) 12420-12430.
10. Rettner, C.T., DeLouise, L. A. and D.J. Auerbach, "Effect of incidence kinetic energy and surface coverage on the dissociative chemisorption of oxygen on W (110)," J. Chem. Phys. **85** (1986) 1131-1149.
11. Dyson, A.J. and P.V. Smith, "A molecular dynamics study of the chemisorption of C₂H₂ and CH₃ on the Si(001)-(2 X 1)surface," Surface Science **375** (1997) 45-54.
12. Komanduri, R. and L.M. Raff, "A review on the molecular dynamics simulation of machining at the atomic scale," Proc. Instn. Mech. Engrs. Part B **215** (2001) 1639-1672.
13. Brenner, D. W., Shenderova, O. A., Harrison, J. A., Stuart, S. J., Ni, B., Susan B Sinott, "A second-generation reactive empirical bond order (REBO) potential energy expression for hydrocarbons ", J. Phys.: Condens. Matter **14** (2002) 783-802.
14. Tersoff, J., "Modeling solid-state chemistry: Interatomic potentials for multicomponent systems," Phys. Rev. B. **39** (1989) 5566-5568.

15. Brenner, D. W., "Empirical potential for hydrocarbons for use in simulating the chemical vapor deposition of diamond films," Phys. Rev. B. **42** (1990) 9458-9471.
16. Verlet, L., "Computer "Experiments" on classical fluids. II. equilibrium correlation functions," Phy. Rev. **165** (1968) 201-214.
17. Beeman, D., "Some multistep methods for use in molecular dynamics calculations," J. Comp. Phys. **20** (1976) 130-139.
18. Allen, M.P. and D.J., Tildesley, "Computer simulation of liquids," (Oxford University Press, 1989).
19. Zhou, D., Gruen, D.M., Qin, L.C., McCauley, T.G. and A.R. Krauss, "Control of diamond film microstructure by Ar additions to CH₄/H₂ microwave plasmas," J. Appl. Phys. **84** (1998) 1981-1989.
20. Spitsyn, B. V., Bouilov, L. L. and B.V. Derajaguin, "Vapor growth of diamond on diamond and other surfaces," Journal of Crystal Growth **52** (1981) 219-226.
21. Reinke, P., Kania, P., P.Oelhafen, "Investigation of the nucleation mechanism in bias-enhanced diamond deposition on silicon and molybdenum," Thin Solid Films **270** (1995) 124-129.
22. Kurihara, K., Sasaki, K., Kawarada, M. and Nagaaki Koshino, "High rate synthesis of diamond by dc plasma jet chemical vapor deposition," Appl. Phys. Lett. **52** (1988) 437-438.
23. Cappelli, C.A. and P.H. Paul, "An investigation of diamond film deposition in a premixed oxyacetylene flame," J. Appl. Phys. **67** (1990) 2596-2602.

24. DeVries, R.C., "Synthesis of diamond under metastable conditions," Ann. Rev. Mater. Sci. **17** (1987) 161-187.
25. Philip, J., Hess, P., Feygelson, T., Butler, J.E., Chattopadhyay, S., chen, K.H. and L.C. Chen, "Elastic, mechanical, and thermal properties of nanocrystalline diamond films," J. Appl. Phys. **93** (2003) 2164 - 2171.
26. Sekaric, L., Parpia, J.M., Craighead, H.G., Feygelson, T., Houston, B.H. and J.E. Butler, "Nanomechanical resonant structures in nanocrystalline diamond," Appl. Phys. Lett. **81** (2002) 4455-4457.
27. Gruen, D. M., Liu, S., Krauss, A.R., Luo, J. and X. Pan, "Fullerenes as precursors for diamond film growth without hydrogen or oxygen additions," Appl. Phys. Lett. **64** (1994) 1502-1504.
28. Zhou, D., McCauley, T.G., Qin, L.C., Krauss, A.R. and D.M. Gruen, "Synthesis of nanocrystalline diamond thin films from Ar-CH₄ microwave plasma" J. Appl. Phys. **83** (1998) 540-543.
29. Goyette, A. N., Lawler, J. E., Anderson, L. W., Gruen, D. M., McCauley, T. G., Zhou, D. and A. R. Krauss, "Spectroscopic determination of carbon dimer densities in Ar-H₂-CH₄ and Ar-H₂-C₆₀ plasmas," J. Phys. D: Appl. Phys. **31** (1998) 1975-1985.
30. Gruen, D. M., Liu, S., Krauss, A.R. and X. Pan, "Buckyball microwave plasmas: Fragmentation and diamond-film growth," J. Appl. Phys. **75** (1994) 1758-1763.
31. Alfonso, D. R., Ulloa, S. E. and D. W. Brenner, "Hydrocarbon adsorption on a diamond (100) stepped surface," Phys. Rev. B **49** (1994) 4948-4953.

32. Alfonso, D. R. and S. E. Ulloa, "Molecular dynamics simulations of methyl-radical deposition on diamond (100) surfaces," Phys. Rev. B **48** (1993) 12235-12239.
33. Hu, Y. and S. B. Sinnott, "Molecular dynamics simulation of thin film nucleation through molecular cluster beam deposition: Effect of incident angle," Nucl. Instr. and Meth. in Phys. Res. B **195** (2002) 329-338.
34. Huang, Z., Pan, Z. Y., Wang, Y. X. and A. J. Du, "Deposition of hydrocarbon molecules on diamond (001) surfaces: atomic scale modeling," Surface and Coating Technology **158-159** (2002) 94-98.
35. Perry M. D. and L. M. Raff, "Theoretical studies of elementary chemisorption reactions on an activated diamond (111) terrace," J. Phys. Chem. **98** (1994) 8128-8133.
36. Perry M. D. and L. M. Raff, "Theoretical studies of elementary chemisorption reactions on an activated diamond (111) ledge surface," J. Phys. Chem. **98** (1994) 4375-4381.
37. Izumi, S., Sato, Y., Hara, S. and S. Sakai, "Development of molecular dynamics potential for Si-H systems and its application to CVD reaction processes," Surface Science **560** (2004) 1-11.
38. Zhu, W., Pan, Z., Ho, Y. and Z. Man, "Impact-induced chemisorption of C₂H₂ on diamond (001) surfaces: a molecular dynamics simulation," Nucl. Instr. and Meth. in Phys. Res. B **153** (1999) 213-217.

39. Hansen, D. A. and J. B. Hudson, "Oxygen scattering and initial chemisorption probability on Ge(100)," Surface Science **254** (1991) 222-234.
40. Belsky, P.; Streiter, R.; Wolf, H. and T. Gessner, "Application of molecular dynamics to the simulation of IPVD,"
41. Vattuone, L., Burghaus, U., Valbusa, U. and M. Rocca, "Breakdown of normal energy scaling at high impact energy for O₂ on Ag(001)," Surface Science **408** (1998) L693-L697.
42. Huang, Z., Pan, Z.Y., Zhua, W.J., Wang, Y.X., and A.J. Du, "Energy dependence of methyl-radical adsorption on diamond (001) – (2 X 1) surface," Surface and Coatings Technology **141** (2001) 246-251.
43. Neyts, E., Bogaerts, A., Gijbels, R., Benedikta, J. and M.C.M. Van de Saden, "Molecular dynamics simulation of the impact behaviour of various hydrocarbon species on DLC," Phys. Rev. B **228** (2005) 315-318.
44. Pailthroe, B. A., "Molecular dynamics simulations of atomic processes at the low temperature diamond(111) surface," J. Appl. Phys. **70** (1991) 543-547.
45. Natale, C. D., Proietti, E, Diamanti, R. and A. D'Amico, "Modeling of APCVD-Doped silicon dioxide deposition process by a modular neural network," IEEE transactions on semiconductor manufacturing, **12** (1999) 109-115.

46. Nami, Z., Misman, O., Erbil, A. and G. S. May, "Semi-empirical neural network modeling of metal-organic chemical vapor deposition," IEEE transactions on semiconductor manufacturing, **10** (1997) 288-294.
47. Bhatikar, S. R. and R. L. Mahajan, "Artificial neural-network-based diagnosis of CVD barrel reactor," IEEE transactions on semiconductor manufacturing, **15** (2002) 71-78.
48. Han, S. and G. S. May, "Using neural network process models to perform PECVD silicon dioxide recipe synthesis via genetic algorithms," IEEE transactions on semiconductor manufacturing, **10** (1997) 279-287.
49. Geisler, J. P., Lee, C.S.G. and G. S. May, "Neurofuzzy modeling of chemical vapor deposition process," IEEE transactions on semiconductor manufacturing, **13** (2000) 46-60.
50. Lorenz, S. Groß, A. and M. Scheffler, "Representing high-dimensional potential-energy surfaces for reactions at surfaces by neural networks," Chemical Physics Letters **395** (2004) 210-215.
51. Hobday, S., Smith, R. and J. Belbruno, "Applications of neural networks to fitting interatomic potential functions," Modelling Simul. Mater. Sci. Eng. **7** (1999) 397-412.
52. Raff, L.M., Malshe, M., Hagan, M., Doughan, D.I., Rockley, M.G. and Komanduri, R., "Ab initio potential-energy surfaces for complex, multichannel systems using modified novelty sampling and feedforward neural networks," J. Chem. Phys. **122** (2005) 084104.

53. Sumpter, B. G. and D. W. Noid, "Potential energy surfaces for macromolecules. A neural network technique," Chemical Physics Letters **192** (1992) 455-462.
54. Sumpter, B. G., Getino, C. and D. W. Noid, "A neural network approach to the study of internal energy flow in molecular systems," J. Chem. Phys. **97** (1992) 293-306.
55. Berendsen, H.J.C., Postma, J.P.M., Gunsteren, W.F.V., DiNola, A. and J.R. Hoak, "Molecular dynamics with coupling to an external bath," J. Chem. Phys. **81** (1984) 3684-3690.
56. Rice, B. M., NoorBatcha, I., Thompson, D.L. and L.M. Raff, "The dynamics dissociative chemisorption of H₂ on a Si(111) surface," J. Chem. Phys. **86** (1987) 1608-1615.
57. Moore, W.J., "Basic Physical Chemistry," [PRENTICE-HALL, INC., Englewood Cliffs, New Jersey 07632, 1983].
58. Raff, L.M. and D. L. Thompson, "The classical trajectory approach to reactive scattering. In theory of chemical reaction dynamics," Baer, M., Ed.; CRC Press: Boca Raton, FL, 1985; Vol III.
59. McCulloch, W.S. and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," Bulletin of Mathematical Biophysics. **5** (1943) 115-133.
60. Farley, B. and W.A. Clark, "Simulation of self-organizing systems by digital computer," IRE Transaction of Information Theory. **4** (1954) 76-84.
61. Rosenblatt, F., "The Perceptron: A probabilistic model for information storage and organization in the brain," Psych. Review. **65** (1958) 386-408.

62. Windrow, B. and M.E. Hoff, "Adaptive Switching circuits," Institute of Radio Engineers, Western electronic Show and Convention, Convention Record, Part **4** (1960) 96-104.
63. Amari, S., "A theory of adaptive pattern classifiers," IEEE Trans. on Electronic Computers. EC-**16** (1967) 299-307.
64. Webros, P., "Advanced forecasting methods for global crisis warning and models of intelligence," General Systems Yearbook (1977).
65. Fukushima, K., Miyake S. and T. Ito, "Neocognitron: a neural network model for a mechanism of visual pattern recognition," IEEE Transactions on Systems, Man, and Cybernetics. SMC -**13** (1975) 826-834.
66. Klopff, A.H., "A drive-reinforcement model of single neuron function: An alternative to the Hebbian neuronal model," AIP Conference Proceedings. **51** (1986) 265-270.
67. McClelland, W.S., Rumelhart, O.E. and the PDP Research Group. "Parallel distributed processing: explorations on the microstructures of cognitron, Vol. **11**," Psychological and Biological Models, MIT Press, Cambridge, MA.
68. Hopfield, J.J. and D. Tank, "Computing with neural circuits; A model," Science. **233** (1986) 625-633.
69. Hagan, M. T., Demuth, H.B. and M.H. Beale, "Neural network design," (Martin Hagan, 2002).
70. Komanduri, R., Chandrasekaran, N. and L. M. Raff, "M.D. Simulation of nanometric cutting of single crystal aluminum-effect of crystal orientation and direction of cutting," Wear **242** (2000) 60-88.

71. Komanduri, R., Chandrasekaran, N. and L. M. Raff, "Effect of tool geometry in nanometric cutting: a molecular dynamics simulation approach," *Wear* **219** (1998) 84-97.
72. Farkas, D., Swygenhoven, H. V. and P. M. Derlet, "Intergranular fracture in nanocrystalline metals," *Phys. Rev. B* **66** (2002) 060101.
73. Xie, J.Q., Feng, J.Y. and H. W. Lu, "Molecular dynamics simulation of low-temperature growth of silicon films by cluster deposition," *Modelling Simul. Mater. Sci. Eng.* **7** (1999) 289-295.
74. Guan, P., McKenzie, D. R. and B. A. Palithorpe, "MD simulations of Ag film growth using the Lennard – Jones potential," *J. Phys. Condens. Matter* **8** (1996) 8753-8762.
75. Shimizu, J., Eda, H., Yoritsune, M. and Etsuji Ohmura, "Molecular dynamics simulation of friction on the atomic scale," *Nanotechnology* **9** (1998) 118-123.
76. Clare, A. G., Hall, M. M., Korwin-Edson, M. L. and A. H. Goldstein, "Biomolecular characterization of glass surfaces," *J. Phys.: Condens. Matter* **15** (2003) s2365-s2375.

VITA

Abdul Nizam Abdul Samadh

Candidate for the Degree of

Master of Science

Thesis: MOLECULAR DYNAMICS (MD) SIMULATIONS OF CHEMICAL VAPOR DEPOSITION (CVD) OF CARBON DIMER ON A DIAMOND (100) SURFACE AND APPLICATION OF NEURAL NETWORKS (NN) FOR EVENT PROBABILITY PREDICTIONS

Major Field: Mechanical Engineering

Biographical:

Education: Received Bachelor of Engineering degree in Mechanical Engineering from University of Madras, TamilNadu, Chennai, India in May, 2001; completed requirements for the Master of Science degree at Oklahoma State University in July, 2005.

Experience: 1. Graduate research assistant in the department of Mechanical and Aerospace engineering at Oklahoma State University, Stillwater, Oklahoma; May 2003- July 2005.