

# Reactions on Surfaces with Neural Networks

vorgelegt von  
Dipl. Phys. Sönke Lorenz  
Berlin

Von der Fakultät II - Mathematik- und Naturwissenschaften  
der Technischen Universität Berlin  
zur Erlangung des akademischen Grades  
Doktor der Naturwissenschaft  
Dr.rer.nat.  
genehmigte Dissertation

Promotionsausschuss:

Vorsitzender: Prof. Dr. W. Richter

Berichter/Gutachter: Prof. Dr. E. Schöll

Berichter/Gutachter: Prof. Dr. M. Scheffler

Tag der wissenschaftlichen Aussprache: 18. September 2001

Berlin 2001  
D 83



---

## Zusammenfassung der Dissertation:

### “Reactions on Surfaces with Neural Networks”

---

Theoretische Untersuchungen der Dynamik chemischer Reaktionen setzen die Kenntnis der Potentialhyperfläche (potential-energy surface = PES), d.h. der potentiellen Energie als Funktion der Freiheitsgrade des betrachteten Systems, voraus. In den letzten Jahren ist es möglich geworden, PESs mit Programmen basierend auf der Dichtefunktionaltheorie (DFT) zu untersuchen. *Ab initio* Rechnungen sind jedoch sehr zeitintensiv und liefern daher nur einen beschränkten, diskreten Satz von Energien. Aufgrund der statistischen Natur des Dissoziationsprozesses kann es allerdings erforderlich sein,  $10^6 - 10^7$  verschiedenen Konfigurationen des Molekül-Oberflächensystems zu berechnen. Das Ziel der Arbeit war die Entwicklung einer neuen, alternativen Methode zur *Interpolation* von *ab initio* Energien unter Verwendung von Neuronalen Netzen (NN) und deren Anwendung in Molekulardynamik-Berechnungen (MD) von Reaktionswahrscheinlichkeiten. Besonderes Interesse galt dabei der Untersuchung der Vergiftung der Wasserstoffdissoziation auf Pd(100) Oberflächen durch Adsorbate wie Schwefel und Kalium. Ein Verständnis solcher Prozesse ist von hoher technologischer Relevanz, für, z.B., die Herstellung neuer Katalysatoren.

Die Interpolationsfähigkeit von Neuronalen Netzen wurde zunächst anhand von sechs-dimensionalen *analytischen* Potentialhyperflächen für die Systeme  $\text{H}_2/\text{Pd}(100)$  und  $\text{H}_2/\text{S}(2 \times 2)/\text{Pd}(100)$  getestet. Dies erlaubte uns zum Einen den Einfluß der Auswahl der zu interpolierenden Energien ausgiebig zu studieren. Zum Anderen ermöglichte dies einen Vergleich der Ergebnisse von MD-Simulationen auf der analytischen und der neuronalen PES. Wir zeigten, daß Neuronale Netze Potentialhyperflächen mehrerer Freiheitsgrade flexibel, schnell und verläßlich interpolieren können. Dabei ist es notwendig, im Fit zusätzliche Konfigurationen als die üblicherweise verwendeten hochsymmetrischen Punkte (top, hollow, bridge) einzubeziehen. Die Auswahl der Punkte auf einem äquidistanten Gitter im 6D-Raum erfordert die Berechnung von  $10^4 - 10^5$  Energien. Mit einem effektiven “Sampling” unter expliziter Berücksichtigung der Korrugation der PES sind nur  $10^3 - 10^4$  Datenpunkte nötig.

Als eine Anwendung der Interpolationsmethode mit Neuronalen Netzen auf *ab initio* Daten wurde im Rahmen dieser Arbeit die PES der Dissoziation von  $\text{H}_2$  auf einer mit Kalium bedeckten Pd(100) Oberfläche mit der *full-potential linear augmented-plane-wave* (FP-LAPW) Methode berechnet. Die Rechnungen zeigten, daß Kalium die Pd(100) Oberfläche - wie auch Schwefel - bezüglich der Dissoziationsrate von  $\text{H}_2$  durch die Bildung von Energiebarrieren verringert. Die Korrugation aufgrund der Adsorption von K-Atomen ist im Vergleich zu Schwefel Adsorbaten deutlich geringer ausgeprägt. Die berechneten 659 *ab initio* Energien wurden mit Neuronalen Netzen gefittet. Die Ergebnisse von MD-Rechnungen auf der NN-PES für die  $\text{K}(2 \times 2)/\text{Pd}(100)$  Oberfläche wurden mit einer bereits existierenden Studie für  $\text{S}(2 \times 2)/\text{Pd}(100)$  verglichen. Die Ergebnisse unterstrichen die Wichtigkeit hoch-dimensionaler Simulationen zur Vorhersage von Adsorptionswahrscheinlichkeiten.

Durch eine Anwendung der Interpolation mit Neuronalen Netzen auf *ab initio* Daten eines anderen Prozesses neben der Dissoziation, der Photodesorption von CO auf  $\text{Cr}_2\text{O}_3(0001)$ , konnte die Flexibilität der vorgestellten Methode demonstriert werden.



---

# Contents

---

<b>1</b>	<b>Introduction</b>	<b>5</b>
<b>I</b>	<b>Theoretical Background</b>	<b>11</b>
<b>2</b>	<b>Adsorption of hydrogen on metal surfaces</b>	<b>13</b>
2.1	Potential-energy surfaces . . . . .	14
2.2	Sticking probability . . . . .	15
2.3	Dissociative adsorption of H <sub>2</sub> on metal surfaces . . . . .	17
2.3.1	Dissociation on transition metal surfaces . . . . .	17
2.3.2	Dissociation on an adsorbate covered metal surface . . . . .	22
<b>3</b>	<b><i>Ab initio</i> total energy calculations and the FP-LAPW method</b>	<b>25</b>
3.1	Density-functional theory . . . . .	25
3.1.1	Exchange-Correlation functional . . . . .	26
3.1.2	Kohn-Sham equations . . . . .	27
3.2	Full potential linear augmented plane wave method FP-LAPW . . . . .	28
3.2.1	APW and the LAPW method . . . . .	29
3.2.2	FP-LAPW method . . . . .	30
3.2.3	Atomic forces . . . . .	31
3.2.4	Super-cell approach . . . . .	31
3.2.5	Brillouin zone integration . . . . .	33
3.3	Mapping of potential-energy surfaces . . . . .	33
<b>4</b>	<b>Interpolation of potential-energy surfaces</b>	<b>35</b>
4.1	Interpolation schemes . . . . .	36
4.2	H <sub>2</sub> /Pd(100)-PES: Interpolation with analytical functions . . . . .	37
4.3	H <sub>2</sub> /Pd(100)-PES: Interpolation with a tight-binding scheme . . . . .	38
4.4	Advantages and disadvantages of various interpolation schemes . . . . .	38
<b>5</b>	<b>Neural Networks</b>	<b>41</b>
5.1	Inspiration from Neuro-science . . . . .	42
5.2	Structure of feed-forward neural networks . . . . .	43
5.3	Function approximation and neural networks . . . . .	45
5.4	Activation functions . . . . .	45
5.5	Supervised Learning - parameter optimisation . . . . .	46
5.6	Optimisation algorithms . . . . .	48
5.6.1	Basics of gradient descent . . . . .	48
5.6.2	Backpropagation . . . . .	49

5.6.3	Conjugate Gradients	50
5.6.4	Newton and Quasi-Newton methods	51
5.6.5	Extended Kalman filter	51
5.7	Generalisation	55
5.8	Regularisation	56
5.9	Normalising the data	56
5.10	Weight initialisation	57
5.11	Ill-conditioning	58
5.12	Fortran program	58
<b>6</b>	<b><i>Ab initio</i> molecular dynamics</b>	<b>61</b>
6.1	Quantum dynamics	62
6.2	Classical dynamics	63
<b>II</b>	<b>Tests: Neural Network fits to analytical functions</b>	<b>65</b>
<b>7</b>	<b>Neural Network tests: Simple analytical functions</b>	<b>67</b>
7.1	Trigonometric function (1D)	67
7.2	Harmonic Oscillator (1D)	70
7.3	Trigonometric function (2D)	71
<b>8</b>	<b>Neural Network test: 6-D analytical PES for H<sub>2</sub>/Pd(100)</b>	<b>73</b>
8.1	<i>Ab initio</i> and analytical PES	74
8.2	Tests of the Neural Network structure	75
8.2.1	Optimisation algorithms	75
8.2.2	Activation functions	79
8.2.3	Ill-conditioning	82
8.2.4	Incorporation of the symmetry	83
8.2.5	Optimised neural network structure	84
8.3	Neural Network PES	84
8.3.1	Neural Network fit of a 2D elbow plot	85
8.3.2	Neural Network fit based on high-symmetric configurations	85
8.3.3	Neural Network fit based on an enhanced lateral grid	88
8.3.4	Neural Network fit based on a dense grid of configurations	89
<b>9</b>	<b>Neural Network test: 6-D analytical PES for H<sub>2</sub>/(2×2)S/Pd(100)</b>	<b>91</b>
9.1	<i>Ab initio</i> and analytical PES	92
9.2	Incorporation of the symmetry	93
9.3	Neural Network PES	93
9.3.1	Neural Network fit based on a dense grid of configurations	94
9.3.2	Neural Network fit based on eleven elbow plots	95
9.3.3	Neural Network fit based on eleven elbow plots & corrugation	96
9.4	Constrained Neural Network fitting approach	98
9.5	Energetic corrugation and its influence on adsorption probabilities	104

---

<b>III Applications: Neural Network fits to <i>ab initio</i> data</b>	<b>109</b>
<b>10 Hydrogen dissociation on the K(2×2)/Pd(100) surface</b>	<b>111</b>
10.1 Hydrogen on K/Pd surfaces in experiments . . . . .	112
10.2 <i>Ab initio</i> potential-energy surface . . . . .	112
10.2.1 Computational details . . . . .	112
10.2.2 Potential-energy surface . . . . .	113
10.3 Incorporation of the symmetry . . . . .	125
10.4 Neural Network PES . . . . .	126
10.4.1 Constrained fitting approach . . . . .	128
10.5 Molecular dynamics with neural networks . . . . .	130
10.5.1 H <sub>2</sub> dissociation on K/Pd(100) versus S/Pd(100) . . . . .	131
10.5.2 Dissociation and scattering of H <sub>2</sub> on K(2×2)/Pd(100) . . . . .	133
<b>11 Photodesorption of CO from Cr<sub>2</sub>O<sub>3</sub>(0001)</b>	<b>137</b>
11.1 Laser induced desorption experiment . . . . .	138
11.2 <i>Ab initio</i> molecular dynamics with Neural Networks . . . . .	140
<b>12 Summary</b>	<b>143</b>
<b>IV Appendix</b>	<b>147</b>
<b>A Derivation of Backpropagation</b>	<b>149</b>
<b>B Extended Kalman filter equations</b>	<b>151</b>
B.1 Kalman filter equations with weighting . . . . .	151
B.2 Kalman filter equations with forces . . . . .	154
<b>C Neural network derivatives</b>	<b>157</b>
C.1 First derivatives with respect to the weights . . . . .	157
C.2 First derivatives with respect to the inputs . . . . .	159
C.3 Second derivatives with respect to the weights and inputs . . . . .	161
C.4 Derivatives of the transfer functions . . . . .	162
<b>D Sampling of the <i>ab initio</i> PES for H<sub>2</sub>/(2×2)K/Pd(100)</b>	<b>165</b>
<b>E Neural Network PES for H<sub>2</sub>/(2×2)K/Pd(100)</b>	<b>169</b>
<b>F Abbreviations</b>	<b>173</b>





---

# Chapter 1

## Introduction

---

Surfaces play a prominent role in our everyday life. A better knowledge of them is vital for the high-technology world we are living in and for future progress. An understanding of surfaces will provide us with important insight into the properties of materials, their production and their growth, into how to protect them from damage or rust and how to protect the environment from hazardous products. The “evolution” from the micro world of today into the nano world of tomorrow will be impossible without a deeper knowledge about surfaces and their interaction with their surroundings.

Processes on surfaces play a crucial technological role, e.g. in the performance of advanced materials. Many chemical reactions are in fact promoted tremendously if they take place on a surface that acts as a catalyst [1]. However, since these processes are often rather complex, in surface science one tries to analyse them focusing first on single steps of the complicated reaction [2]. In this work we will concentrate on dissociative adsorption processes since they constitute the first step in heterogenous catalysis. Furthermore, they are often the rate-limiting step, the bottleneck of the reaction as for example in ammonia synthesis or CO oxidation [3]. In particular we are interested in the poisoning (reduction) and promotion (enhancement) of a surface reaction by adsorbates.

Catalysts are not only used to increase the output of a chemical reaction but also to convert hazardous waste into less harmful products. The most prominent example is the car exhaust catalyst. Its reactivity is, e.g., poisoned by the presence of sulphur in petrol [4]. The technological relevance of catalysis is reflected by the fact that throughout the world more than 90% of the chemical manufacturing processes utilise catalysts in one form or another [5]. In spite of their importance, the lack of understanding of molecular and atomistic processes at surfaces is significant. The reason are the different length and time scales of such reactions, where one has to bridge many orders of magnitude - from a single dissociation event taking around 10 femtoseconds to a whole concert of processes lasting several micro- or even milliseconds [6, 7].

In the last decade there had been enormous progress concerning the accurate and reliable description of the interaction of atoms and molecules with surfaces. This progress was largely due to the improvement of programs based on density functional theory (DFT) [8, 9, 10, 11]. Yet, these calculations are still computational demanding. We will present a combined DFT and neural network approach to study surface reaction rates - a method inspired by nature and now used in various research areas [12, 13, 14].

An understanding of the underlying mechanisms of poisoning and promotion by adsorbates and their consequences on reaction rates is of strong technological relevance for, e.g., designing better catalysts. A theoretical description of such reactions requires a knowledge of the potential energy of the system as a function of the coordinates of the reactants, the so-called *potential-energy surface* (PES). The PES is the central quantity in any theoretical description as it rules all chemical processes.

The dissociative adsorption of hydrogen on metal surfaces has become a model system for theoretical and experimental investigations of molecular-surface interactions. Detailed reviews of the subject are available by Kroes [15] and Groß [16]. It is an important step in many catalytic reactions. In recent years it has been possible to map out potential-energy surfaces of H<sub>2</sub> dissociation on metal surfaces with DFT based computer programs [17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28]. These studies have largely concentrated on the variation of the activity on clean surfaces of different substrate materials. Also the influence of co-adsorbates on the surface reactivity for different processes has been addressed by DFT methods [29, 30, 31, 32, 33]. For H<sub>2</sub> dissociation, experiments showed that Pd(100) surfaces are poisoned not only by S adatoms [4], but also in the presence of a K adlayer [34]. This is of special interest, since commonly it has been argued that electronegative atoms like sulphur poison and electropositive alkali atoms like potassium promote a catalytic reaction [35, 36]. Theoretically it has been found that H<sub>2</sub> is able to dissociate freely on the Pd(100) surface [22], whereas due to the presence of sulphur and also potassium energy barriers form which poison the process [30, 33, 37].

However, up to now a theoretical comparison of the dynamical consequences of the adsorption of an electronegative adsorbate and an electropositive on the dissociation of hydrogen on the same substrate is missing. This is due to the fact that for a realistic description of the performance of a catalyst in action a knowledge about the distribution of energy barriers to the reaction is not sufficient. One has to go beyond the DFT calculations and extend the simulation to different time scales. A number of methods have been developed for this second step, each one applies to different situations as summarised in Tab. 1.1.

An understanding of the dissociation of molecules on surfaces requires on the one hand to study the motion of the involved particles on an atomistic scale. On the other, for a proper description of real catalysts, a good statistical average over many such events is necessary. Molecular dynamics simulations (MD) allow us to bridge the time gap and to extend the DFT calculations to processes lasting up to several picoseconds, covering the time scale we are interested in, and even nano- or microseconds [38]. In MD calculations the atoms are allowed to move under the influence of the forces as described by Newton's equations of motion or the Schrödinger equation. In *ab initio* molecular dynamics the potential and the forces are determined by DFT. *Ab initio* MD simulations are accurate, but time-consuming. Of particular importance in studies of dissociation dynamics on surfaces is the sticking probability, i.e. the probability that an incoming molecule dissociates and then the atoms adsorb at the surface. For its calculation we need to consider a good statistical average over the different initial configurations of the molecule approaching the surface with a certain energy. We need, e.g., 1000 trajectories per energy. A single event might take up to 1 ps divided into one

Method	Type of information	Time scale	Length scale
Density-functional theory	Microscopic	-	$\leq 10^3$ atoms
<i>Ab initio</i> molecular dynamics	Microscopic	$t \leq 10$ ps	$\leq 10^2$ atoms
Semi-empirical molecular dynamics	Microscopic	$t \leq 1$ ns	$\leq 10^3$ atoms
Kinetic Monte Carlo	Micro-/Mesoscopic	$1 \text{ ps} \leq t \leq 1 \text{ ns}$	$\leq 1 \mu\text{m}$
Rate equations	Averaged	$0.1 \text{ s} \leq t \leq \infty$	All
Continuum Equations	Macroscopic	$1 \text{ s} \leq t \leq \infty$	$\geq 10 \text{ nm}$

Table 1.1: *The time and length scales handled by different theoretical approaches to study chemical reactions. We will concentrate in this work on the first two rows of this table, i.e. density functional theory calculations and ab initio molecular dynamics.*

time step every 10 fs by the numerical integration of the equations of motion. Hence, a theoretical simulation of a sticking curve might require to calculate

	10	kinetic energies
x	1.000	trajectories / kinetic energy
x	1	pico seconds / trajectory
x	100	time steps / pico second
=	$10^6$	time steps (!),

i.e.  $10^6$  evaluations of the potential and the forces. This is a formidable task. *Ab initio* molecular dynamics are limited to dynamical studies of only a few trajectories [39] and do not allow the direct determination of reaction probabilities.

Therefore Groß and Scheffler proposed a “divide and conquer” approach for *ab initio* molecular dynamics which is built on three independent steps [40]: first, one determines the *ab initio* PES by DFT. In a second step an interpolation between the actually calculated *ab initio* energies is performed. The last step consists of a molecular dynamics calculation on this continuous representation of the *ab initio* PES. The crucial part of this approach is the interpolation of the total energies. The interaction of a diatomic molecule with a well-defined surface is at least six-dimensional, corresponding to the six degrees of freedom of the molecule and a fixed substrate. The latter assumption is often fulfilled for densely packed metal surfaces. For example on Si(100), the rearrangement upon adsorption is indeed crucial for the adsorption and desorption mechanism [39] and we easily arrive at 12 and more dimensions.

Six-dimensional *ab initio* dynamics calculations based on an analytical interpolation of total energies showed that dynamical effects can be significant and differences from a static theory remarkable (see Kroes [15] and references therein). They have advanced the understanding of the dissociation dynamics greatly and caused the modification of established concepts. Some phenomena, like the so-called *steering effect* [41], can only be modelled in a theoretical simulation including a sufficiently large number of degrees

of freedom. Thus, high-dimensional dynamical studies lead to progress not only in the quantitative, but also in the qualitative understanding of processes on surfaces.

However, the fitting of the *ab initio* energies to a continuous representation is a highly non-trivial task. A high-dimensional, flexible, accurate, reliable and fast interpolation scheme is needed. Ideally this method should be general to allow its application to a wide range of problems. Various approaches to fit a PES can be found in the literature [42, 43, 44, 45, 46, 47, 48, 28, 27]. All of the proposed methods have some advantages and some drawbacks. For instance, the fitting of *ab initio* data using analytical functions [41, 49, 50, 51] requires an appropriate choice of an analytical form, which is very cumbersome to find in high dimensions. It is therefore fair to say that despite its importance a general, fast and accurate interpolation tool for potential-energy surfaces is still lacking.

As an alternative to the hitherto proposed fitting schemes we will introduce an interpolation method based on neural networks [52, 53, 54]. Neural Networks can in principle approximate any continuous function to arbitrary accuracy [55, 56]. They do not require any assumptions about the functional form of the underlying problem. Neural networks are general, i.e. the same computer program can be used for different problems. The network *learns* itself what the best configuration is. Furthermore, the evaluation of the potential energy with a fitted neural network is cheap and the derivatives, the forces, are obtainable. Therefore, provided that the number of parameters and required data for a good fit scale favourably with dimension, neural networks will be ideal for molecular dynamics applications. The main area of research in neural computing is devoted to classification or pattern recognition problems which is a profoundly different task from the one we are interested in, the interpolation of a multidimensional function. However, recently neural networks have also been applied to the interpolation of potential-energy surfaces [57, 58, 59, 60, 61]. These works have concentrated on low-dimensional studies of the PES of molecules in the gas phase. We want to extend these applications to reactions of molecules on surfaces on a high-dimensional potential-energy surface and employ the neural network PES in molecular dynamics simulations.

This work is devoted to the development of a new interpolation method for *ab initio* data and its application to the poisoning of catalytic surfaces by adsorbates. We have studied the reduction of the hydrogen dissociation rate due to the presence of potassium atoms on a Pd(100) surface using density functional theory, neural networks and molecular dynamics. We will present a comparison of the poisoning effect due to the co-adsorption of an electropositive atom like potassium to an existing study with an electronegative adsorbate like sulphur [33, 51, 62] on the same substrate, i.e. the Pd(100) surface.

In order to learn more about the practical approximation ability of neural networks it is essential to find realistic test problems. Analytical potential-energy surfaces provide ideal test cases for various reasons. They are fast to evaluate and therefore allow us to study the influence of the data sampling on the quality of the neural network fit in great detail. Furthermore, they have been successfully used for the *ab initio* description of the hydrogen dissociation on metal surfaces using a six-dimensional PES [26, 41, 49, 63, 64, 51]. Moreover, as an additional check of the accuracy of the obtained neural network model we are able to compare the neural network MD results to calculations

---

performed on the analytical PES. We have chosen analytical PESs for the clean [41] as well as the sulphur covered [33] Pd(100) surface as test problems.

The remainder of this work is divided into three parts: a description of the theoretical background (Chapter 2-6), the presentation of the neural network test results (Chapter 7-9) and the application of the neural network model to the interpolation of *ab initio* total energies (Chapter 10 & 11). In Chapter 2 we briefly describe the general concepts in theoretical studies of hydrogen adsorption on metal surfaces. Chapter 3 consists of the theoretical framework of density functional theory. Following this we give an overview over the existing interpolation methods for *ab initio* total energies (Chapter 4). In Chapter 5 we will then introduce Neural Networks as an alternative interpolation approach for potential-energy surfaces. The theoretical part ends with the concepts of molecular dynamics in Chapter 6.

We have tested the neural network approach first with simple low-dimensional analytical functions as described in Chapter 7. Chapter 8 and Chapter 9 refer to the results of the high-dimensional test problems, the analytical PES for hydrogen dissociation on the clean and the sulphur covered Pd(100) surface, respectively.

In Chapter 10 we present the application of the “divide and conquer” approach to *ab initio* molecular dynamics for the dissociation of hydrogen on the potassium covered Pd(100) surface using neural networks. We have first determined the *ab initio* potential-energy surface by density functional theory. In a second step, we have interpolated the *ab initio* data with neural networks to a continuous representation. We have then applied the neural network PES in classical molecular dynamics simulations and determined the reaction rate for dissociative adsorption. In addition, we compared this results to the dynamics on the sulphur covered Pd(100) surface.

Another important part of a catalytic reaction is the desorption of a molecule from a surface. In order to demonstrate the flexibility of the neural network approach we present in Chapter 11 the neural network interpolation of *ab initio* data for the photodesorption of CO from Cr<sub>2</sub>O<sub>3</sub>(0001). The neural network PES has been employed in molecular dynamics simulations of the rotational alignment of the desorbing molecule. A comparison of the theoretical results to experimental data is given. Chapter 12 summarises the results of this work.



**Part I.**  
**Theoretical Background**





---

## Chapter 2

### Adsorption of hydrogen on metal surfaces

---

Understanding reactions on surfaces plays an important role in a wide range of technologically relevant applications, among those “heterogenous catalysis”. A “catalyst” is a substance that enhances the rate at which a certain chemical is produced. “Heterogenous” refers to the situation that the reaction occurs when reactants and catalyst are present in different phases, e.g. in the gas and the solid phase. The presence of a catalytic surface, which is exposed to the reactants, is crucial for the reaction: Adsorbed particles on the surface encounter an energy barrier which is lower than that of the chemical reaction between the same reactants in the gas phase. However, since these processes are often rather complex, one tries to analyse them focusing first on single steps of the complicated reaction. Processes relevant to heterogeneous catalysis are, e.g., the adsorption of atoms and molecules, often dissociatively, on a solid surface from the gas phase, the diffusion on the surface, chemical reactions of different adparticles on the surface and the desorption of the reactant and the product. The rate limiting step, e.g., in the ammonia synthesis or CO oxidation, is often the dissociation of molecules on surfaces. Much theoretical and experimental work is devoted to the measurement and calculation of the adsorption or sticking probability of molecules on surfaces, i.e. the percentage of molecules impinging on a surface which are not scattered back into the gas phase.

Hydrogen dissociation on metal surfaces has become the model system for bond-breaking processes. Hydrogen is the simplest molecule which makes it a feasible candidate for theoretical investigations. At the same time hydrogen is also well-suited for performing experiments which allows a fruitful interaction between theory and experiment.

In the following we will briefly describe the general concepts in theoretical studies of adsorption on surfaces. Having introduced the potential-energy surface as the central quantity in theoretical descriptions of chemical reactions we will define the sticking probability of adsorption events. In the last part of this chapter we will give a short overview about research in the field of dissociative adsorption of hydrogen on metal surfaces. We will describe the potential-energy surface of the  $\text{H}_2$  dissociation on the clean palladium surface and the dynamics of the  $\text{H}_2$ -surface interaction that proceeds on it. We will also embark on the influence of adsorbates on the dissociation process. Adsorbates can act as a promoter or poisoner of reactions on substrates. For a more detailed description of hydrogen dissociation please refer to ref. [65, 16].

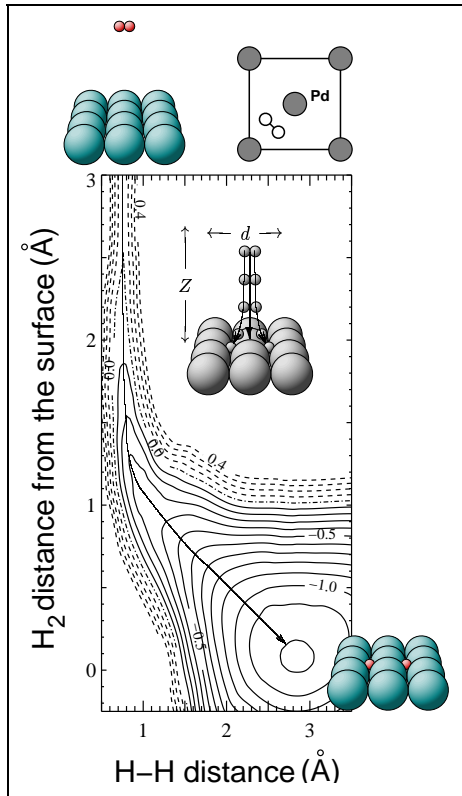
## 2.1 Potential-energy surfaces

For dynamical studies of dissociation processes we need the energy for many configurations of the nuclei, i.e. we need the potential-energy as a function of the coordinates of the reactants, the so-called *potential-energy surface (PES)*:

$$\text{PES:} \quad \equiv \quad V(\{R_i^j\}),$$

where  $R_i^j$  is the coordinate of nuclei  $i$  of species  $j$  taking part in the reaction. Potential-energy surfaces are multidimensional functions, their dimensionality depends in principle on the number of reactants involved.

The PES is the central quantity in theoretical studies of chemical reactions. It is usually viewed in two-dimensional cuts through the multi-dimensional configuration space, the so-called *elbow plots*, in the  $Zd$  plane, where  $Z$  is the distance of the hydrogen molecule from the surface and  $d$  is the distance between the two hydrogen atoms, the bond length.



**Figure 2.1:** Contour plot of the ab initio PES along a 2D cut through the 6D space of  $\text{H}_2/\text{Pd}(100)$ , the so-called elbow plot, from [29]. Energies are in eV/ $\text{H}_2$  molecule. Inset: schematic drawing of the dissociation process. Top right part:  $\text{H}_2$  orientation on the surface.

As an example the elbow plot for the dissociation of  $\text{H}_2/\text{Pd}(100)$  over the bridge site [29,22,41] has been plotted in Fig. 2.1. In the top-left part of the elbow-plot the molecule is far away from the surface in the gas phase, defined as energy-zero. In the bottom-right corner of the plot the hydrogen molecule is already dissociated, the two hydrogen atoms are adsorbed on the surface. The solid line between these two parts marks the path towards dissociation in this configuration. As we can see from Fig. 2.1 there is a path from the top left to the bottom right along which the energy monotonously decreases. The dissociation is not only exothermic, there is also no barrier to dissociation, i.e., the molecule can freely dissociate. Such a dissociation path is called *non-activated*. In contrast, paths with an energy barrier to dissociation are called *activated*, as for example in the system  $\text{H}_2/\text{Cu}(111)$  [18].

An important concept in theoretical studies of dissociation processes is to find the *reaction path* from the molecule in the gas phase to the adsorbed atoms on the surface. The reaction path is usually defined as the minimum energy path towards dissociation. In the case of hydrogen on the  $\text{Pd}(100)$  surface this path is indeed the one plotted in Fig. 2.1. However, it is important to notice that this is just one two-dimensional cut through the multi-dimensional configuration space. Other configurations may exist, which do have an energy barrier for dissociation.

## 2.2 Sticking probability

The sticking probability is defined as the fraction of particles that remain on the surface and are not scattered back into the gas phase. This quantity can be measured in beam experiments, where a beam of particles is directed onto a crystal surface. The particles are either atoms or molecules. In the case of molecules, the process can be subdivided into molecular and dissociative adsorption. Here molecular adsorption refers to the situation where the molecule stays intact on the surface, while in the latter the molecule breaks up and its fragments stay on the surface.

There is one fundamental difference between atomic and molecular adsorption on the one side and dissociative adsorption on the other. In atomic or molecular adsorption the key process is the transfer of kinetic energy to the surface. For theoretical descriptions of atomic or molecular adsorption it is therefore crucial to consider energy dissipation of the impinging particle to the substrate. Let  $P_E(\epsilon)$  be the probability that an incoming particle with kinetic energy  $E$  will transfer energy  $\epsilon$  to the surface. The atomic or molecular sticking probability can then be expressed as

$$S(E) = \int_E^{\infty} d\epsilon P_E(\epsilon) . \quad (2.1)$$

It corresponds to the fraction of particles that transfer more energy to the surface than their initial energy. This excess energy has to be transferred to substrate excitations like electron-hole pairs or phonons. With increasing kinetic energy this dissipation process becomes less and less efficient, the fraction of particles that lose more than their initial kinetic energy becomes smaller at higher energies. Therefore the adsorption probability decreases with increasing kinetic energy. The sticking probability as a function of the kinetic energy of the impinging particles (or equivalently as a function of the beam temperature) can be seen in Fig. 2.2 (from the textbook by Zangwill [2]). A sticking probability of 1 means that all particles which approach the crystal stay on the surface, a value of 0 reflects that all particles are scattered back into the gas phase.

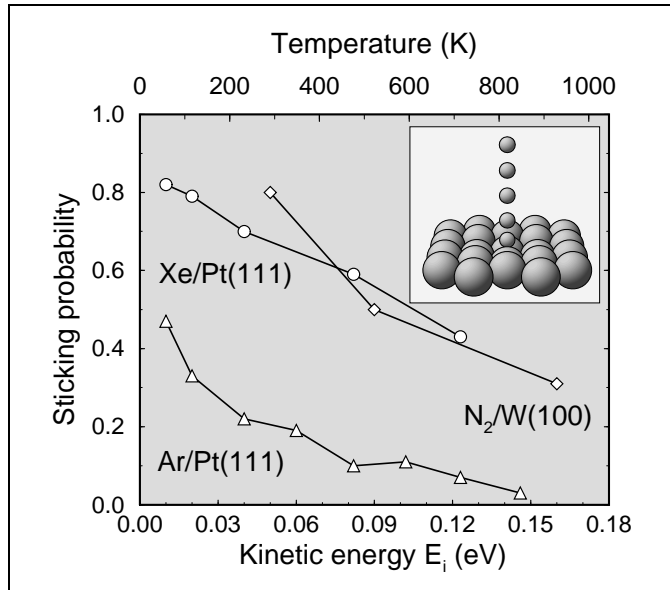


Figure 2.2: Atomic sticking probability versus kinetic energy for Xe and Ar on Pt(111) and molecular sticking probability of N<sub>2</sub>/W(100). These examples are taken from the textbook by Zangwill [2]. Inset: schematic representation of the adsorption process.

Now in the case of dissociative adsorption another process of energy transfer comes into play. The kinetic energy of the impinging molecule can be transferred into translational energy of the atomic fragments relative to each other. Eventually they will also dissipate their kinetic energy and come to rest at the surface. However, especially for light molecules like hydrogen dissociating on metal surfaces the energy transfer to the substrate is very small due to the large mass mismatch of the reactants. The sticking process is almost entirely determined by the bond-breaking of the atoms in the molecule. If furthermore no substantial surface rearrangement upon adsorption occurs, as it is usually the case for light molecules dissociating on close-packed metal surfaces, then the new and important channel for energy transfer makes it possible to neglect the surface degrees of freedom.

For the dissociation of hydrogen molecules on metal surfaces we are therefore able to describe this process with a potential energy depending only on the six degrees of freedom of the molecule. However, we should keep in mind that for heavier molecules and also for the dissociation on semiconductor surfaces the rearrangement of the surface atoms can be important and needs to be incorporated explicitly. The dimensionality of the potential-energy surface will then be even higher than six.

Figure 2.3 shows the experimental sticking probability versus kinetic energy for the dissociation process of hydrogen on a Cu(111) surface. In this system the dissociation is hindered by an energy barrier. Sticking can only occur, if the kinetic energy of the impinging molecule is large enough to overcome the barrier towards dissociation. Even at high kinetic energies the adsorption probability is significantly reduced in comparison to the atomic and molecular adsorption in Fig. 2.2 (please note the different energy and sticking ranges in Fig. 2.2 and Fig. 2.3, respectively). However, dissociative adsorption probabilities can differ by orders of magnitude for different systems. In the case of hydrogen molecules on many transition metal surfaces the sticking at room temperature is about 0.5 [66, 4], whereas for  $\text{H}_2/\text{Si}$  it is only  $10^{-8}$  [67], and for  $\text{N}_2/\text{Ru}$  it is even  $10^{-13}$  [68].

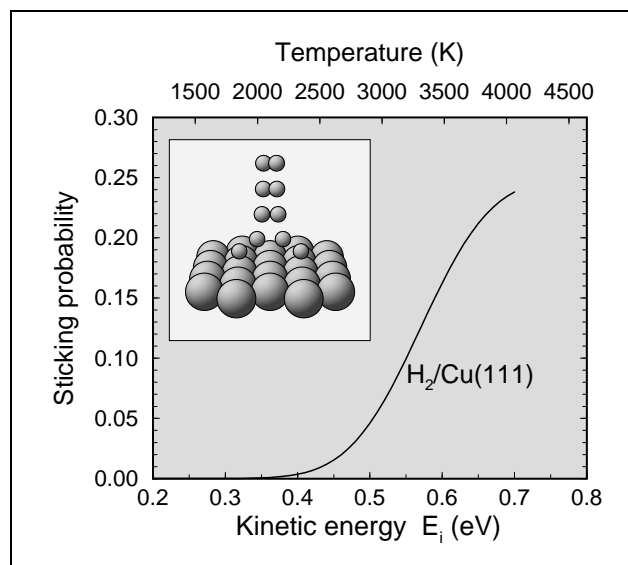


Figure 2.3: Dissociative adsorption probability versus kinetic energy of  $\text{H}_2/\text{Cu}(111)$  (from Time-of-flight measurements by Rettner, Michelsen, and Auerbach [69]). The curve shown has been obtained by fitting the adsorption data while neglecting the effects of rotation. The molecules are initially in the vibrational ground state. Inset: schematic representation of the dissociation process. Compare energy and sticking range with Fig. 2.2 for atomic and molecular adsorption.

## 2.3 Dissociative adsorption of H<sub>2</sub> on metal surfaces

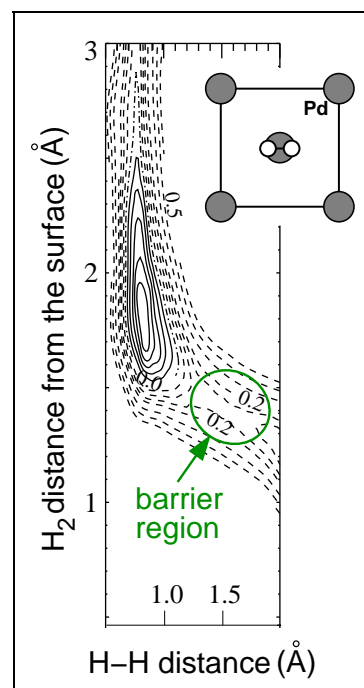
Theoretical studies based on *ab initio* calculations showed that hydrogen dissociation on a noble metal surface like Cu(111) [18] and Cu(100) [19] is activated, the molecule exhibits a barrier for the bond breaking process. The PES for both surfaces shows a strong corrugation. These investigations have established that even on metal surfaces, where the electronic density in front of the surface is rather smeared out, the bond-breaking process is a very localised process close to the surface. It involves the hybridisation of molecular orbitals with certain metal states, in particular the d-states which are spatially strongly varying. It is the chemical nature of the dissociation that leads to a strong corrugation of the potential energy.

Dissociative adsorption is a dynamical process. As we will lay out in the following, a good treatment of the dynamics is indeed crucial to the understanding of reactions on surfaces. But first we will briefly describe some general features of PES for the bond-breaking of hydrogen on transition metals. For a more detailed discussion we refer to Scheffler and Stampfl [70], Groß [40] and Kroes [63].

### 2.3.1 Dissociation on transition metal surfaces

Transition metal surfaces are usually very reactive with respect to the dissociation of hydrogen, like for example Pd(100) [4], Pd(111) [71, 48, 28], W(100) [72] and Ni(110) [4, 27]. Here the term “reactivity” refers to the surfaces ability to break bonds of an approaching molecule and to adsorb the fragments, which is often the rate limiting step in catalytic reactions.

A good knowledge of the high-dimensional PES of the molecule surface system is mandatory for a detailed understanding of the reaction, as the PES rules the scattering and the dissociation. Wilke *et al.* have performed density-functional theory calculations (DFT) of the interaction of H<sub>2</sub> with a Pd(100) transition metal surface [29, 22]. They found that the process is non-activated, i.e. the minimum pathway towards dissociation - the bridge site between the Pd atoms - shows no energetic barrier. However, if we look at another cut through the configuration space in the same system, we find that paths exist, in fact the majority of them, which have an energy barrier as, e.g., the one plotted in Fig. 2.4. Here, where the molecule approaches the surfaces at the on-top site, a barrier of approximately 0.15 eV exists. It is obvious from Fig. 2.1 and Fig. 2.4 that neglecting the high dimensionality, i.e., assuming that the elbows for different choices of  $(X_c, Y_c, \theta, \phi)$  are similar, is by no means justified.



**Figure 2.4:** Contour plot of the *ab initio* PES for hydrogen dissociation on a Pd(100) surface (from [29]). Energies are in eV per H<sub>2</sub> molecule. The contour spacing is 0.05 eV.

This shows that typically several pathways exist along which a molecule may dissociate and the associated energy barriers may be different: Some pathways may be activated, some may not. For example, going through the periodic table from Rh to Pd to Ag it was found [73] that for:

⇒ Rh: most pathways are not hindered by a barrier,

⇒ Pd: most pathways are hindered,

⇒ Ag: all pathways have an energy barrier.

Thus, these three transition metals cover the range from a nearly completely unactivated system [Rh(100)] where the system gains energy on adsorption (exothermic) over a surface exhibiting activated as well as unactivated pathways for H<sub>2</sub> dissociation [Pd(100)], but still being exothermic, to a surface where adsorption is always hindered by an energy barrier and endothermic [Ag(100)] [73]. Yet, it is not only important that there *is* a barrier towards dissociative adsorption but also *where* the barrier is located. On the Rh surface the few barriers are situated in the entrance channel, i.e. when the bond length of the molecule is not significantly stretched and the centre of mass is still far away from the surface [73]. For the Ag surface the lowest barrier is found very close to the surface and at a H-H distance which is by about 100% stretched compared to the free molecule [73]. Thus, the hydrogen bond is nearly broken when the molecule has reached the top of the energy barrier.

Hammer *et al.* showed by analysing the wave functions of the H<sub>2</sub>-surface system at the location of the barrier that the differences between the different metals should be described in a covalent picture [18, 20] (see also the earlier study by Hjelmberg *et al.* (1979) for H<sub>2</sub> at jellium [74]). As illustrated in Fig 2.5, at the barrier the H<sub>2</sub>-surface interaction has already produced a splitting into states which are bonding between the molecule and the substrate and ones which are antibonding. Let us consider a substrate from the middle of the transition-metal series, e.g. Ru or Rh, with the Fermi-level in the middle of the *d*-band in panel (a) of Fig. 2.5. This implies that the resonances  $\sigma_g$  and  $\sigma_u$  in panels (c) and (d) are filled with electrons. These states are on the one hand bonding with respect to the molecule-substrate interaction, their filling implies an attraction of the molecule to the surface. On the other hand, the filling of the  $\sigma_u$  resonance weakens the H-H bond. Thus, the molecules are strongly attracted to the surface and at the same time the molecular bond is weakened. Yet, if the substrate Fermi level is well above the *d*-band, as for a noble metal like Ag, also the antibonding molecule-surface states become filled. This results in a repulsive interaction between the molecule and the substrate and leads to the formation of an energy barrier.

With this view the differences in the chemical activity of transition metals can be easily explained. Ag is chemically inert because its Fermi level is about 3 eV above the top of the *d*-band resulting in a repulsive molecule-substrate interaction. Its left neighbour in the periodic table of elements, Pd, is chemically more active since its Fermi-level lies within the *d*-band as plotted in Fig. 2.5. The elements further to the left, namely Rh and Ru, are even more so, because their Fermi-levels are closer to the middle of the *d*-band.

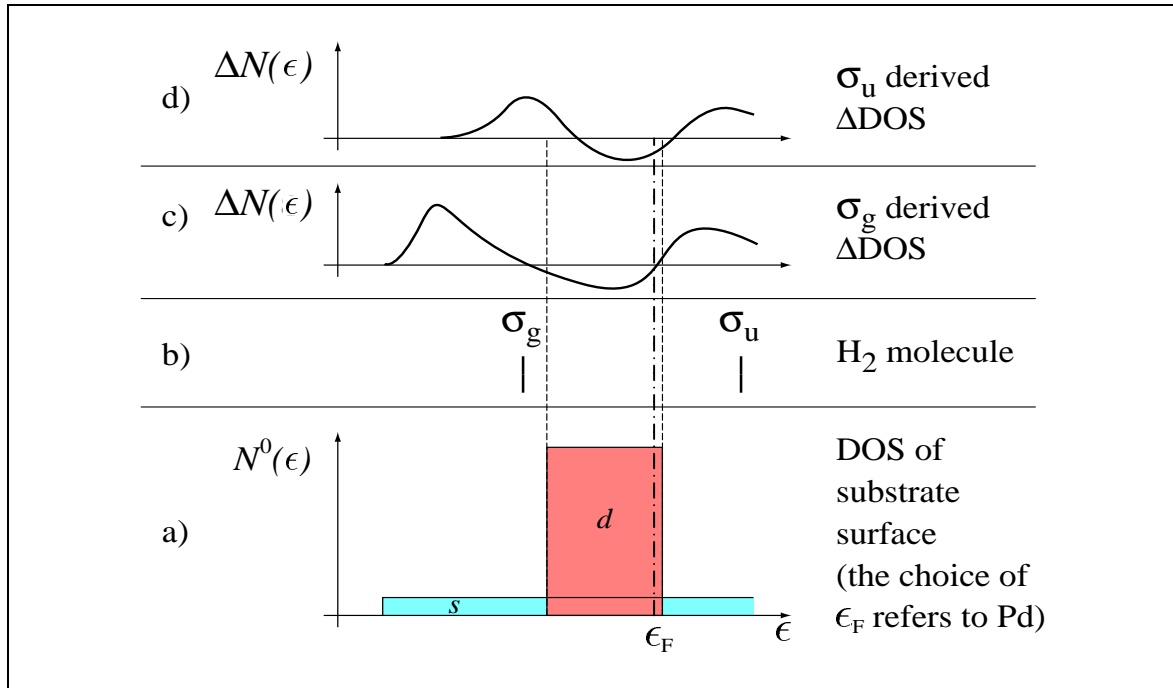


Figure 2.5: Schematic description of the interaction of H<sub>2</sub> at the energy barrier towards dissociative adsorption at transition-metal surfaces. The bottom panel (a) shows the density of states (DOS) for a transition metal before adsorption, the panel (b) shows the energy levels of a free H<sub>2</sub> molecule: the bonding state  $\sigma_g$  is filled with two electrons and the antibonding state  $\sigma_u$  is empty. The interaction between the H<sub>2</sub>  $\sigma_g$ -level and the substrate  $s$ - and  $d$ -bands give rise to a broadening and the formations of an antibonding level at about the upper edge of the  $d$ -band and a bonding level below the  $d$ -band, see panel (c). Panel (d) shows that the interaction between the H<sub>2</sub>  $\sigma_u$ -level with the substrate  $s$ - and  $d$ -bands gives rise to a broadening and the formation of a bonding level (at about the lower edge of the  $d$ -band) and an antibonding level (above the  $d$  band). From Scheffler and Stampfl [70].

### Dynamics of H<sub>2</sub> dissociation at transition-metal surfaces

To obtain a full description of a dynamical process like the dissociation of hydrogen on transition metal surfaces it is necessary to go beyond the DFT calculations and to perform molecular dynamics simulations of the molecules approaching the surface (see e.g. Groß et al. [41]). We will demonstrate in the following, that sometimes dynamical effects can indeed be significant and differences from a static theory noticeable.

We have seen that the dissociative adsorption of hydrogen at Pd(100) can proceed without an energy barrier. However, we recall that the majority of pathways has a barrier for the reaction. This co-existence of non-activated and activated pathways to dissociation has important dynamical consequences. An analytical representation of the *ab initio* PES has been used for a quantum dynamical study in which all six hydrogen degrees of freedom were taken into account explicitly while the substrate was kept fixed [41]. The results are plotted in Fig. 2.6 together with the calculated adsorption

probability for Rh(100) [73] and experimental beam data for hydrogen dissociation on Pd(100) [75]. One unexpected result of Fig. 2.6 is that for low kinetic energies with  $E_i \leq 0.05$  eV the probability for sticking on the Pd and the Rh substrates are very similar. This is surprising since the PES of Rh exhibits many pathways with vanishing energy barrier towards dissociation whereas Pd has only few. In fact, both substrates give a sticking value as high as 75 % for low  $E_i$  corresponding to typical thermal kinetic energies. But while for Rh the sticking probability always remains high, it decreases for Pd to about 25 %. The calculated sticking probability for Pd(100) agrees with the experimental results semi-quantitatively and reproduces the general trend, namely the initial decrease of sticking followed by an increase at higher kinetic energies. It was believed, that the initial decrease of the sticking probability at low kinetic energies corresponds to a situation in which molecular adsorption is involved, cf. Fig. 2.2. In the so-called *precursor* model the molecule would not directly dissociate, but is first trapped molecularly in a precursor state before it dissociates. This trapping process would then be less efficient at higher kinetic energies and thus results in the initial drop of the sticking probability.

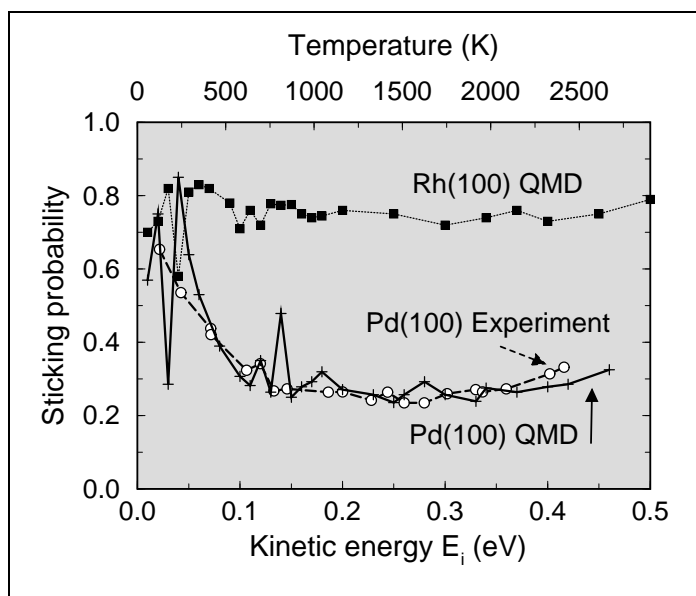


Figure 2.6: Sticking probability for a  $H_2$  beam under normal incidence on a Pd(100) and Rh(100) surface. Theory: 6-d quantum molecular dynamics (QMD) with  $H_2$  molecules in their rotational and vibrational ground state. QMD for Rh and Pd by Eichler et al. [73], molecular beam experiment by Rendulic et al. [75].

However, the *ab initio* calculations did not find a molecular adsorption well, i.e. no precursor state. Furthermore, in the quantum dynamical calculation no energy transfer from the molecule to the substrate is considered. The reason for the initial decrease is a purely dynamical effect, the so-called *steering effect*. Although the majority of pathways exhibits energy barriers to dissociation with a rather broad distribution of heights and positions, slow molecules can be very efficiently steered to the non-activated sites on the surface and dissociate. This process is illustrated in Fig. 2.7 (from [16]). For both parts of the plot the initially non-vibrating and non-rotating molecule impinges on the surface under the same conditions, except for the kinetic energy. Far away from the surface both molecules are oriented almost perpendicular to the surface, a position in which they would not be able to dissociate. Yet, due to the anisotropy of the PES, forces can reorient the molecule to a parallel configuration. The slow molecule in Fig. 2.7(a)



reaches this orientation and dissociates, whereas the molecule with the higher kinetic energy in Fig. 2.7(b) is too fast to be fully-reoriented, it hits the repulsive wall of the PES before it can dissociate and is scattered back into the gas phase. Hence, it is a purely dynamic effect which leads to a high sticking probability at low kinetic energies. At higher energies the steering is less efficient causing the drop of the sticking curve. At even higher kinetic energies the dissociation follows the typical scenario for activated dissociative adsorption, the molecules simply have enough energy to overcome the barriers, compare with Fig. 2.3.

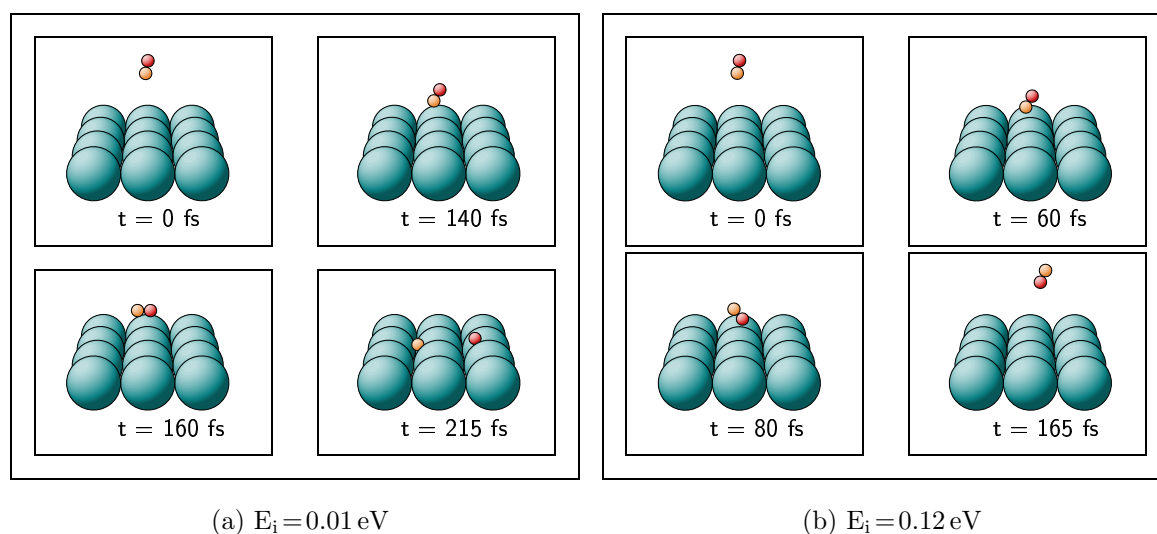


Figure 2.7: *Steering effect. Snapshots of ab initio molecular dynamics trajectories for H<sub>2</sub> molecules on Pd(100). The molecules are initially not vibrating and not rotating. For both trajectories the initial conditions are the same, except for the kinetic energy. A molecule with a low kinetic energy (a) can be steered towards more favourable geometries and dissociates. With higher  $E_i$  (b) the “steering” effect is less efficient, the molecule hits the repulsive part of the PES and is scattered back. From Groß [16].*

These studies established the importance of high-dimensional dynamical calculations for dissociative adsorption reactions. In a static picture as discussed before, the high reactivity of Pd at low kinetic energy cannot be understood. The efficiency of steering depends on the speed of the incoming molecule and on the shape of the PES. Therefore, to evaluate the sticking probability as a measure of the surface reactivity, it is important to consider all degrees of freedom of the incoming particle. Neglecting any of them in lower-dimensional studies would lead to a steering which is much less efficient or even not present at all. Thus high-dimensional simulations not only lead to progress in the quantitative, but also in the qualitative understanding of processes on surfaces. We note, as much as “steering” is important to understand the high reactivity of Pd at low  $E_i$ , for other systems, which on the grounds of the electronic structure alone may be expected to exhibit a high reactivity, an “anti-steering” may occur, which drives approaching particles not toward the best configuration but rather against an energy barrier.

### 2.3.2 Dissociation on an adsorbate covered metal surface

Adsorbates can have a strong influence on the reactivity of a particular surface. They can act as a promoter or a poisoner of the reaction, they can enhance or decrease the reaction rate. Such additives are of great technological importance, e.g. in designing better catalysts. It is well known that the activity of car-exhaust catalysts is significantly reduced by the presence of lead, but also sulphur poisons the catalyst.

It is generally the case that electropositive adsorbates like alkali atoms tend to act as promoters, while electronegative species like sulphur are poisons [1]. In the ammonia synthesis, for instance, alkali atoms like Cs and Na promote the dissociation of  $N_2$  on Ru-based catalysts, whereas S poisons the reaction [31]. However, this simple picture breaks down for hydrogen dissociation. Potassium is indeed a promoter for the dissociation on Cu(110), yet on a number of other surfaces it acts as a poisoner, see table 2.1. Electronegative sulphur poisons the hydrogen dissociation on Pd(100) [75], but oxygen is found to act as a promoter on Ni(111) up to a coverage of one eighth of a monolayer. Poisoning and promoting effects of adsorbates have been explained by an ensemble of mechanisms, for an overview we refer, e.g., to Bird *et al.* [36]. They range from direct and indirect chemical interactions, the modification of the local density of states, especially at the Fermi-level, and electrostatic effects like dipole-dipole interactions. Yet, in this work we will focus on the dynamical consequences of the promoting or poisoning effect of an adsorbate only.

Promoter			Poisoner		
Molecule	Adsorbate	Surface	Molecule	Adsorbate	Surface
$N_2$	Cs	Ru(0001) [31]	$N_2$	S	Ru(0001) [31]
$N_2$	Na	Ru(0001) [31]	CO	S	Ni(100) [1]
$N_2$	K	Fe(111) [76]	$H_2$	K	W(100) [77]
CO	K	Pd(100) [1]	$H_2$	K	Pd(111) [78]
CO	K	Ni(100) [1]	$H_2$	K	Pt(111) [79]
$H_2$	K	Cu(110) [80]	$H_2$	K	Fe(100) [76]
$H_2$	O ( $\Theta < 1/8$ )	Ni(111) [81]	$H_2$	K	Ni(111) [81]

Table 2.1: Promotion and poisoning of molecular and dissociative adsorption rates of molecules by adsorbates ( $\Theta$  refers to the coverage of the adsorbate on the surface).

#### $H_2$ dissociation over S( $2 \times 2$ )/Pd(100)

For the Pd(100) surface it is experimentally well known that sulphur adsorbates lead to a reduction of the hydrogen dissociation probability. At a kinetic energy of  $E_i = 0.05$  eV the dissociation probability at the clean surface is about 60%, at the sulphur covered surface it decreases to a value below 1% [4]. Density functional theory calculations have shown that the dissociation of hydrogen on the sulphur covered Pd(100) surface is activated, i.e. the dissociation is hindered by energy barriers [33]. In Fig. 2.8 three elbow

plots on the six-dimensional PES are plotted, where the molecular axis is kept parallel to the surface. Due to the presence of the sulphur atoms the dissociation over the

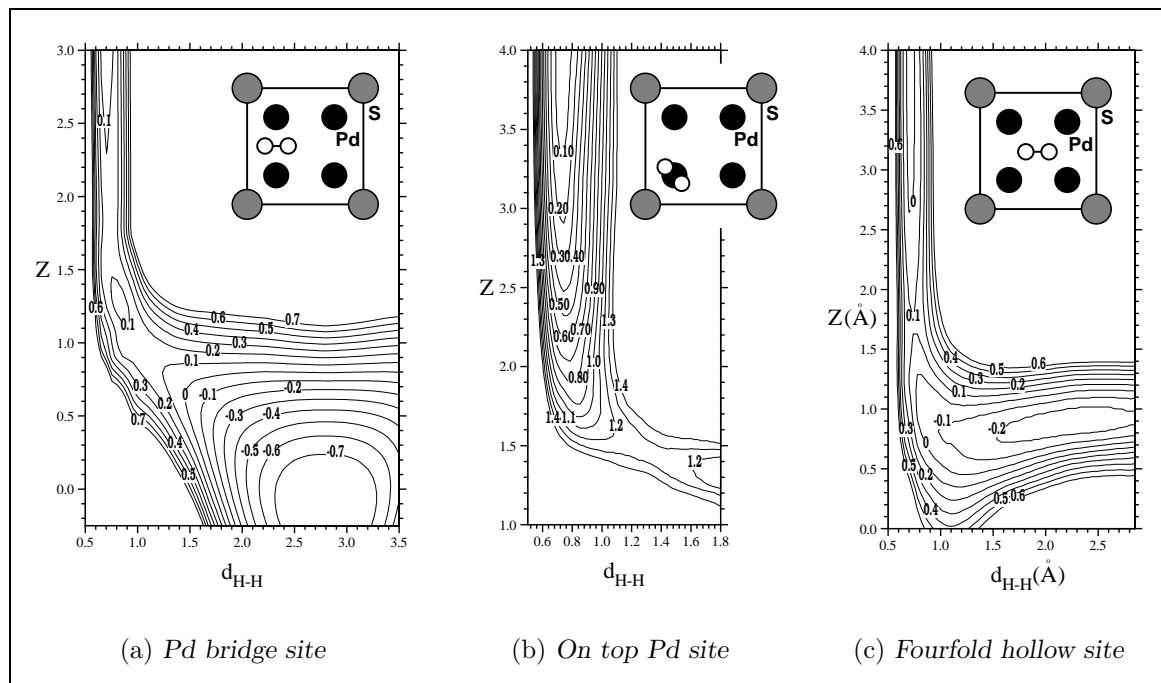


Figure 2.8: Contour plots through the six-dimensional *ab initio* PES of H<sub>2</sub> dissociation over (2×2)S/Pd(100). Insets: geometries of the dissociation pathways. Contour lines in eV/H<sub>2</sub> molecule. From Wei et al. [33].

bridge site between the palladium atoms exhibits now a barrier of 0.16 eV (Fig. 2.8(a)), whereas this site was non-activated on the clean Pd surface (Fig. 2.1). The reaction is indeed poisoned by the adsorbate on the surface. Furthermore, the PES is now highly corrugated. If we look at the dissociation over the on top site on a palladium atom, the barrier rises from 0.15 eV over the clean surface (Fig. 2.4) to 1.3 eV on the sulphur-covered surface (Fig. 2.8(b)). The minimum pathway towards dissociation is now the fourfold hollow site, where the distance of the hydrogen molecule from the sulphur atoms is at its maximum, see Fig. 2.8(c). The energy barrier in this configuration is 0.11 eV. The building up of a barrier is related to an indirect interaction with the sulphur atoms. They cause a downshift of the Pd d-bands at the surface which leads to the population of anti-bonding molecule-surface states [33].

An analytical fit to the *ab initio* PES has been performed and employed in quantum mechanical and classical dynamical calculations [51]. The results are plotted in Fig. 2.9. The theoretical results are significantly larger than the experimental results, which has been proposed to be due to the presence of subsurface sulphur in the experimental samples. Furthermore it has been found that steering is also operative on the activated potential-energy surface. Moreover, Fig. 2.9 shows that the classical molecular dynamics calculations over-estimate the sticking compared to quantum results. This suppression is a consequence of the strong corrugation and anisotropy of the PES which gives rise to large zero-point energies. This energy needs to be taken from the incident beam and

therefore the effective kinetic energy of the impinging molecule is reduced. Consequently the fraction of the molecules which are able to overcome a certain barrier towards dissociation is reduced and the sticking probability is lower. However, the overall trend of the sticking curve with increasing kinetic energy is well reproduced also by classical dynamics.

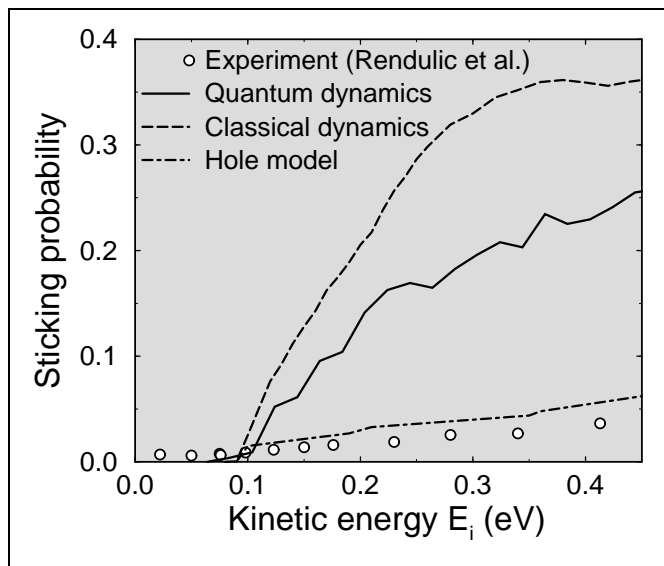


Figure 2.9: Sticking probability versus kinetic energy for a  $H_2$  beam under normal incidence on a  $(2 \times 2)S/Pd(100)$  surface. Theory: six-dimensional quantum and classical dynamics on an analytical PES by Groß et al. [51]. Experiment: from [4].

## $H_2$ dissociation over $K(2 \times 2)/Pd(100)$

It has been found experimentally that the Pd(100) surface is poisoned for hydrogen dissociation in the presence of a K adlayer [34]. At a potassium coverage of  $\Theta = 0.25$  the sticking probability is 30 times less than that on the clean surface. The influence of potassium adatoms on the dissociative adsorption of hydrogen on Pd(100) has also been addressed theoretically by Wilke and Cohen [37]. The calculations showed that indeed potassium adsorbates hinder hydrogen dissociation by forming energy barriers. In comparison to the sulphur-covered Pd(100) surface, the corrugation is less strong in the presence of potassium adsorbates. The authors report that changes of the electrostatic potential, the modification of the electronic states at the surface, and changes in the local density of states near the Fermi-level are simultaneously present. All appear to play a role at the various stages of the dissociation.

However, Wilke and Cohen focused in their work on the system  $(2 \times 2)K/Pd(100)$  only on two dissociation pathways, the fourfold hollow site and the potassium bridge site. The information they collected is not sufficient for a study of the dynamical consequences of the poisoning effect with a weaker corrugation as in the case with sulphur adsorbates. A comparison of the sticking probability for  $H_2$  dissociation on a metal surface like Pd(100) with an electronegative adsorbate like sulphur and an electropositive adsorbate, as for instance potassium, is still lacking. Since the dissociation event is a dynamical process and is often the rate limiting step in heterogenous catalysis, we will re-address this issue and extend the *ab initio* calculations of Wilke and Cohen [37] to various adsorption sites.

---

## Chapter 3

### *Ab initio* total energy calculations and the FP-LAPW method

---

Theoretical studies of reaction dynamics on surfaces, like the dissociative adsorption of diatomic molecules on metal surfaces, require a knowledge of the potential energy of the moving nuclei taking part in the process. *Ab initio* total energy calculations have proven to be a powerful and accurate tool to calculate such properties. Different quantum mechanical methods, like Hartree-Fock, configuration interaction and coupled cluster approaches, are being used in electronic structure theory for decades. In recent years much progress in the description of processes on surfaces has been made through density functional theory and its practical implementations. In the following sections we will briefly describe the theoretical foundations of this framework and its implementation in the WIEN 97-code [82, 83, 84].

### 3.1 Density-functional theory

For a theoretical description of chemical reactions a quantum mechanical many-body problem including all electronic and ionic degrees of freedom needs to be solved. However, for complex systems it is not possible to obtain this solution in closed form. A practical implementation requires the introduction of certain approximations.

In many chemical reactions one can assume that the dynamics of the electrons and the nuclei decouple due to their large mass mismatch: this basically means that the electrons are assumed to follow the movements of the nuclei instantaneously. Whatever the configuration and the dynamics of the nuclei are, the electrons will be in the ground state of that geometry. This is the Born-Oppenheimer-approximation [85], which we consider to be valid throughout the following. This means that in a first step the electronic problem for a particular configuration will be solved, and then in the second step the dynamics of the nuclei have to be considered.

Even with this simplification one has to deal in the many-body Schrödinger equation with a wave function  $\Psi(\{\mathbf{r}_i\})$  depending on all coordinates of the  $N$  electrons of the system. This problem remains formidable. The major achievement of density-functional theory (DFT) is the replacement of the wave function  $\Psi(\{\mathbf{r}_i\})$  by the electron density  $n(\mathbf{r})$  of the  $N$ -electron system as the varying property. The density depends - in contrast

to the multi-dimensional wave function - only on *three* variables, which is an enormous reduction of complexity.

The foundations of DFT are formulated in the Hohenberg-Kohn theorem [86]. The main idea is that physical quantities as expectation values of quantum-mechanical operators are functionals of the electron density. In particular the energy functional of the system is minimised by the ground state electron density  $n_0(\mathbf{r})$  [86]. With this the variational problem of the many-body Schrödinger equation for the electrons can be transformed into a variational problem of an energy functional:

$$E_0 = \langle \Psi_0 | H | \Psi_0 \rangle \leq \langle \Psi | H | \Psi \rangle = E[\Psi] = E[n] , \quad (3.1)$$

with  $E_0$  the energy of the ground state,  $\Psi_0$  the ground state wave function, and  $n(\mathbf{r})$  is the electron density of the  $N$ -electron system.

Starting from here the idea of Kohn and Sham [87] was to map the many-particle problem onto a single particle problem retaining the exact description of the many body effects. They wrote the energy functional  $E[n]$  in the form:

$$E[n] = T_s[n] + U[n] + E_{xc}[n] , \quad (3.2)$$

where

- $T_s[n]$  is the kinetic energy of a system of non-interacting particles,
- $U[n]$  is the average Coulomb-interaction of the electrons:

$$U[n] = \frac{e^2}{8\pi\epsilon_0} \int d\mathbf{r} \int d\mathbf{r}' \frac{n(\mathbf{r})n(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} + \int d\mathbf{r} V_{ext}(\mathbf{r})n(\mathbf{r}) , \quad (3.3)$$

with  $V_{ext}$  being the external potential in which the electrons move, i.e. the Coulomb potential due to the nuclei,

- $E_{xc}[n]$ , the exchange-correlation functional, accounts for the Pauli principle, dynamical correlations due to the Coulomb repulsion, and the correction of the self-interaction present in Eq. 3.3.

### 3.1.1 Exchange-Correlation functional

The exchange-correlation functional  $E_{xc}$  contains the quantum mechanical many-body effects. It is an universal functional, i.e. it is independent of the particular system considered. However, this functional is not known and therefore needs to be approximated. In the local-density approximation (LDA), the non-local exchange correlation energy is locally described by the exchange- and correlation energy density of a homogenous electron gas:

$$LDA : \quad E_{xc}[n] = \int d\mathbf{r} n(\mathbf{r})\epsilon_{xc}(n(\mathbf{r})) . \quad (3.4)$$

The system is composed of small systems with a locally constant density.  $E_{xc}(n)$  can be exactly determined for the homogeneous electron gas. As an example of the LDA

please refer to Perdew and Zunger [88]. Within the LDA lattice constants of solids agree well with experimental values. However, for binding energies of solids and molecules, and most prominently for barriers of chemical reactions [18], the LDA results are not sufficiently accurate.

A natural approach to improve the exchange-correlation functional is to include information about the gradient of the density. This is done in the generalised gradient approximation (GGA) [89, 90, 91, 92, 93]:

$$GGA : \quad E_{xc}[n] = \int d\mathbf{r} n(\mathbf{r}) \epsilon_{xc}(n(\mathbf{r}), \nabla n(\mathbf{r})) . \quad (3.5)$$

Binding energies and energy barriers are clearly improved within the GGA (error  $\leq 0.1-0.2$  eV for many systems [18, 94]). Since we are interested in such quantities, the GGA [91] will be used in the following.

### 3.1.2 Kohn-Sham equations

What remains to be expressed in equation (3.2) is the kinetic energy operator  $T_s[n]$ . This operator is not known explicitly in a closed form, but it can be evaluated exactly as follows, if  $\varphi_i$  are the wave functions of the non-interacting particles:

$$T_s = \sum_i \langle \varphi_i | -\frac{\hbar^2}{2m} \nabla^2 | \varphi_i \rangle . \quad (3.6)$$

The variational problem of the energy functional (3.2) under the constraint of constant particle number ( $N = \int d\mathbf{r} n(\mathbf{r})$ ) leads to:

$$\delta \left[ E[n] - \mu \left( \int d\mathbf{r} n(\mathbf{r}) - N \right) \right] = 0 \quad \implies \quad \frac{\delta E[n]}{\delta n(\mathbf{r})} = \frac{\delta T_s[n]}{\delta n(\mathbf{r})} + V_{eff} = \mu , \quad (3.7)$$

where  $\mu$  is the associated Lagrange multiplier, or the chemical potential. The effective potential  $V_{eff}$  is given by:

$$V_{eff} = V_{ext}(\mathbf{r}) + V_{xc}(\mathbf{r}) + V_H(\mathbf{r}) = V_{ext}(\mathbf{r}) + \frac{\delta E_{xc}}{\delta n(\mathbf{r})} + \frac{e^2}{4\pi\epsilon_0} \int d\mathbf{r}' \frac{n(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} , \quad (3.8)$$

where  $n(\mathbf{r})$  is now the ground-state density of a non-interacting electron system. The density is expressed as a sum over single-particle states with wave functions  $\varphi_i(\mathbf{r})$  and occupation numbers  $f_i$ :

$$n(\mathbf{r}) = \sum_{i=1} f_i |\varphi_i(\mathbf{r})|^2 . \quad (3.9)$$

Since  $T_s(n)$  is the kinetic energy of non-interacting electrons, Eq. 3.7 is solved by:

$$\left[ -\frac{\hbar^2}{2m} \nabla^2 + V_{eff}(\mathbf{r}) \right] \varphi_i(\mathbf{r}) = \epsilon_i \varphi_i(\mathbf{r}) . \quad (3.10)$$

The single-particle states  $\varphi_i(\mathbf{r})$  are obtained by self-consistently solving these effective single-particle equations, the Kohn-Sham equations, together with Eq. (3.8) and Eq. (3.9). Self-consistency means an iterative procedure: from an initial guessed electron density the potential is generated and then the eigenfunctions are obtained as a solution of the Kohn-Sham equations. These eigenfunctions define the new density, which serves mixed together with the old one as the input for the next iteration. This procedure is repeated until convergence of the density and hence the energy is reached.

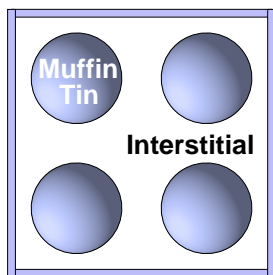
## 3.2 Full potential linear augmented plane wave method FP-LAPW

Different methods have been developed to solve the Kohn-Sham equations (3.10), for instance the *pseudo potential plane wave method* (PPW) [11, 10, 9], the *linear muffin-tin orbital method* (LMTO) [95, 96, 97] and the *linear augmented-plane-waves method* (LAPW) [8, 98]. For a numerical solution the wave functions  $\varphi$  is expanded into a basis set of functions  $\Phi$ :

$$\varphi = \sum_i c_i \Phi_i . \quad (3.11)$$

The methods differ by the choice of the set of basis functions. In the PPW method plane waves are used in the expansion. They have the advantage that they form a complete set, they fulfil Bloch's theorem and they are easy to implement. The potential of the crystal is replaced by a norm-conserving pseudopotential [99], which exploits the fact that the core electrons are hardly involved in binding processes. The influence of the core electrons is represented by an effective potential. This approach reduces the number of electrons that have to be taken into account explicitly. However, for the description of very localised states, like the valence electrons of transition metals, a large number of plane waves have to be applied.

The LMTO and the LAPW method are *all-electron* methods. The phrase *all-electron* refers to the calculation of every electron state in contrast to the PPW method with pseudo wave functions and pseudo states. In both the LMTO and the LAPW method space is divided into spheres - so-called muffin-tins - around each nucleus and an interstitial region in between (Fig. 3.1). This allows the use of different sets of basis functions in the different regions. Within the muffin-tin spheres atomic like basis functions are being used, whereas in the interstitial region the LMTO method applies Hankel & Bessel functions and the LAPW method plane waves. We will now discuss the LAPW method, which we will use in this work, in more detail, starting with its historic precursor, the *augmented plane wave method* (APW).



**Figure 3.1:** Space divided into muffin-tin spheres and an interstitial region



### 3.2.1 APW and the LAPW method

The APW method [100, 101] by Slater was motivated by the fact that near an atomic nucleus the potential and wave functions vary strongly and look rather atomic-like, whereas further away they are much smoother. Therefore Slater divided space into regions - muffin-tins (MT) and interstitial (I) - and used different basis functions in these regions: inside each muffin-tin radial solutions of the Schrödinger equation and plane waves in the interstitial region:

$$\Phi_{\mathbf{G}}^{\text{APW}}(\mathbf{k}, \mathbf{r}) = \begin{cases} e^{i(\mathbf{k}+\mathbf{G})\cdot\mathbf{r}} & : \mathbf{r} \in \text{I} \\ \sum_{lm} A_{lm} u_l(r, \epsilon) Y_{lm}(\mathbf{r}) & : \mathbf{r} \in \text{MT} \end{cases}, \quad (3.12)$$

where  $\mathbf{k}$  is the Bloch vector,  $\mathbf{G}$  the vector of the reciprocal lattice. The coefficients  $A_{lm}$  are determined by the requirement of a continuous wave function at the sphere boundary. The function  $u_l(r, \epsilon)$  is the regular solution of the radial Schrödinger equation:

$$\left[ -\frac{\partial^2}{\partial r^2} + \frac{l(l+1)}{r^2} + V_{00}(r) - \epsilon \right] r u_l(r, \epsilon) = 0, \quad (3.13)$$

where the term  $V_{00}(r)$  is the spherical part of the expansion of the potential into spherical harmonics:

$$V(\mathbf{r}) = \sum_{l=0}^{\infty} \sum_{m=-l}^l V_{lm}(r) Y_{lm}(r) \quad : \mathbf{r} \in \text{MT}. \quad (3.14)$$

The APWs are solutions of the Schrödinger equation inside the sphere at the energy  $\epsilon$ , and only there. Its value must be set equal to the band energy. This explicit energy dependence makes the APWs rather inflexible as the band energy is not known a priori, but one of the quantities sought after. In order to increase the variational freedom of the APWs, to allow for changes in the wave function as the band energy deviates from this reference energy, the linear APW method (LAPW) [102, 103, 8, 98] has been introduced. Its wave function is constructed of the radial function  $u_l$  and an additional term, the energy derivative  $\dot{u}_l(r)$ :

$$\Phi_{\mathbf{G}}^{\text{APW}}(\mathbf{k}, \mathbf{r}) = \begin{cases} e^{i(\mathbf{k}+\mathbf{G})\cdot\mathbf{r}} & : \mathbf{r} \in \text{I} \\ \sum_{lm} [A_{lm} u_l(r) + B_{lm} \dot{u}_l(r)] Y_{lm}(\mathbf{r}) & : \mathbf{r} \in \text{MT} \end{cases}. \quad (3.15)$$

The coefficients  $A_{lm}$  and  $B_{lm}$  are determined by matching the value and the slope of the basis functions at the sphere boundary. For this purpose the plane waves are expanded into Bessel functions  $j_l(|\mathbf{k} + \mathbf{G}| |\mathbf{r}|)$  and spherical harmonics  $Y_{lm}(\mathbf{r})$ :

$$e^{i(\mathbf{k}+\mathbf{G})\cdot\mathbf{r}} = 4\pi \sum_{l=0}^{\infty} \sum_{|m| \leq l} i^l j_l(|\mathbf{k} + \mathbf{G}| |\mathbf{r}|) Y_{lm}^*(\mathbf{k} + \mathbf{G}) Y_{lm}(\mathbf{r}). \quad (3.16)$$

In practice the wave function is expanded only up to a parameter  $l_{max}^{wav}$ .

Having defined the LAPWs (3.16) we can now rewrite the solutions of the Kohn-Sham equations (3.11) as:

$$\varphi_{\mathbf{k}} = \sum_{|\mathbf{G}| < G_{max}^{wav}} c_{\mathbf{k}}^{\mathbf{G}} \Phi_{\mathbf{G}} . \quad (3.17)$$

Together with  $l_{max}^{wav}$ ,  $G_{max}^{wav}$  is another so-called *cut-off parameter* which determines the size of the basis set and hence the quality of the calculation.

The Taylor expansion of the radial function around an energy  $E_l$  introduces an error in the wave function of  $\mathcal{O}([\epsilon - E_l]^2)$ :

$$u_l(r, \epsilon_l) = u_l(r, E_l) + \dot{u}_l(r)(\epsilon - E_l) + \mathcal{O}([\epsilon - E_l]^2) . \quad (3.18)$$

The corresponding error in the energy is of fourth order. Due to this high order error, the LAPW basis functions form a good basis set for a reasonable choice of the energy parameter  $E_l$ .

One difficulty remains: each reference energy can only be chosen once per angular momentum  $l$ . Thus bands with the same quantum number  $l$  but different energies, like 1s- or 2s-states, can not be calculated in one step. In order to circumvent this problem, local orbitals [104], which are completely localised inside the MTs, have been introduced. The idea is to add an additional radial function  $u_l(\mathbf{r}, \tilde{\epsilon}_l)$  at a different linearisation energy  $\tilde{\epsilon}_l$  with coefficients  $C_{lm}$  to the LAPWs (3.16) inside each MT. The coefficients  $C_{lm}$  are calculated by the requirement of a vanishing wave function of these orbitals at the sphere boundary.

### 3.2.2 FP-LAPW method

The APW and the LAPW method use only the spherical part of the potential inside the muffin-tins (see (3.13)) and a constant potential in the interstitial region. In the *full potential* LAPW method no shape approximations are performed. The potential inside each sphere is expanded into spherical harmonics and in the interstitial into plane waves:

$$V(\mathbf{r}) \equiv \begin{cases} \sum_{\mathbf{G}} V_{\mathbf{G}} e^{i\mathbf{G}\cdot\mathbf{r}} & : \mathbf{r} \in \text{I} \\ \sum_{lm} V_{lm}(r) Y_{lm}(r) & : \mathbf{r} \in \text{MT} \end{cases} . \quad (3.19)$$

The representation of the density follows analogous. As in the case of the wave function, the expansions are truncated by the *cut-off parameters*  $G_{max}^{pot}$  for the plane waves in the interstitial region and  $l_{max}^{pot}$ , which limits the number of non-spherical components to the potential in the muffin-tins.

It is advantageous to take account for the symmetry of the system in the calculation. Not all components in the potential and density expansion need to be calculated in one step. The interstitial density has the symmetry of the space group, inside the sphere the density has the site symmetry and the density within atomic spheres related by a symmetry operation are identical, apart from a rotation. This is exploited using the so-called *stars* in the interstitial and *lattice harmonics* within the muffin-tins [104].

### 3.2.3 Atomic forces

For the minimisation of the total energy with respect to the coordinates of the nuclei the atomic forces need to be calculated. The force on atom  $i$  is based on the total derivative of the energy, including implicit dependencies, with respect to the atomic position  $\mathbf{R}_i$ :

$$\mathbf{F}_i = -\frac{\partial E}{\partial \mathbf{R}_i} - \sum_i \frac{\partial E}{\partial \varphi_i} \frac{d\varphi_i}{d\mathbf{R}_i} - \sum_i \frac{\partial E}{\partial \varphi_i^*} \frac{d\varphi_i^*}{d\mathbf{R}_i}. \quad (3.20)$$

Only the first term is of physical nature the others derive from an incomplete basis set. The changes in the electronic wave functions when one ion moves from one position to another contribute to the force on the ion. However, for plane wave basis sets these last two terms in equation (3.20) vanish if each electronic wave function is an eigenstate of the Hamiltonian, i.e. at the end of the self-consistent cycle. In this case the partial derivative of the Kohn-Sham energy with respect to the position of an ion gives the physical force on the ion, which is referred to as the Hellmann-Feynmann theorem [105].

The forces are very sensitive to errors in the wave functions. Therefore accurate forces can only be calculated close to self-consistency. The error in the energies is only second order with respect to errors in the wave function, whereas the error in the forces is first order. Consequently the cost is significantly lower to calculate an accurate total energy than an accurate force.

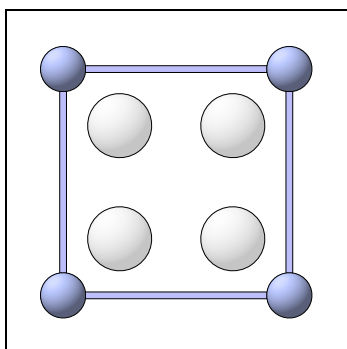
In quantum mechanical calculations with position dependent basis sets, additional terms to the forces appear to represent the derivative of the basis set with respect to the position of the ion (Pulay force [106]). For plane wave basis sets these forces vanish [107], for atomic-like orbitals as used in the LAPW method they need to be considered. In the WIEN97-code the calculation of the forces has been successfully implemented by Kohler et al. [83].

### 3.2.4 Super-cell approach

Density-functional theory allows us to map the many-body problem onto a single-particle problem. In a solid we still have to find the solution for a system of an infinite number of non-interacting particles moving in the static potential of an infinite number of nuclei. However, one can take advantage of the periodicity of a crystal. Plane-wave based methods are ideally suited to treat periodic systems because the plane waves automatically fulfil Bloch's theorem [108] which states that for a periodic system the wave function can be written as a product of a phase factor  $e^{i\mathbf{k}\cdot\mathbf{r}}$  and a lattice periodic function  $w_{\mathbf{k}}(\mathbf{r})$ :

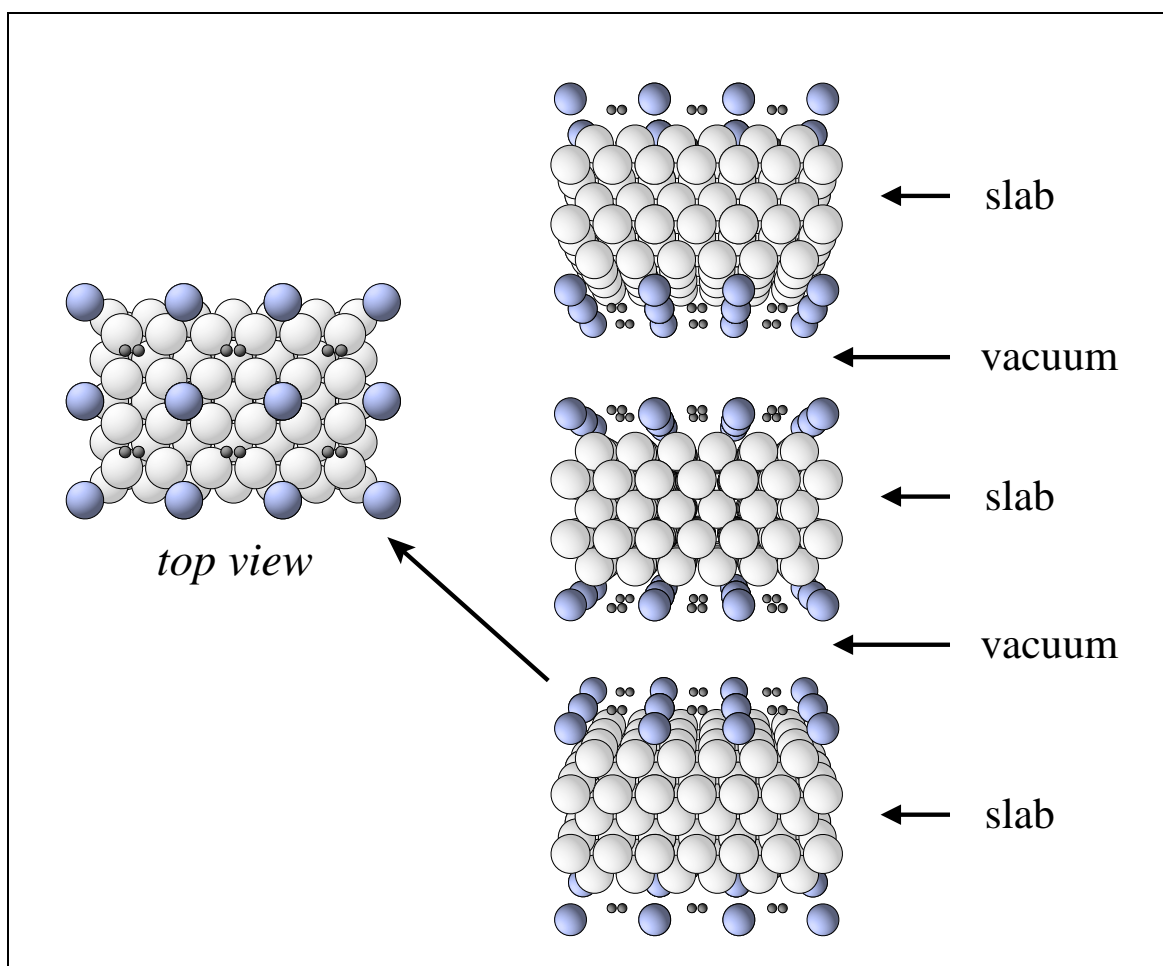
$$\varphi_{\mathbf{k}}(\mathbf{r}) = e^{i\mathbf{k}\cdot\mathbf{r}} w_{\mathbf{k}}(\mathbf{r}). \quad (3.21)$$

In a bulk material Bloch's theorem can be applied due to the three dimensional periodicity of the lattice. We are interested in the study of a dissociation process on a surface, where the translational symmetry of the system normal to the surface is broken. Still it is desirable to exploit the computational efficiency of plane-wave methods. This is achieved by artificially restoring the three-dimensional periodicity.



**Figure 3.2:** The  $(2 \times 2)$  surface unit cell

In the super-cell approach the surface is described by an infinite array of slabs separated by vacuum regions (see Fig. 3.3). The thickness of the each slab is determined by the requirement that the two surfaces of the slab do not interact with each other. The separation between the different slabs has to be chosen large enough to ensure that the slabs are not interacting. If we place an adsorbate on the surface the periodicity in the lateral direction is also broken. In order to recover this periodicity we study ordered adsorbate layers that are described by laterally repeated surface unit cells. The size of the cell depends on the system and problem of interest. In our case we want to calculate the dissociation of hydrogen on a potassium covered palladium surface. The potassium coverage is 25% and this is represented by a  $(2 \times 2)$  unit cell as shown in Fig. 3.2.



**Figure 3.3:** Side view of the super-cell of a 5 mono-layer slab with an adsorbate in a  $p(2 \times 2)$ -unit cell and a diatomic molecule on the surface. Adsorbates are placed on both sides of each slab. Each slab is separated by a vacuum region. A top view of one slab is also indicated.

### 3.2.5 Brillouin zone integration

In order to determine the charge density or the total energy, it is necessary to evaluate sums over occupied states, which for crystals become integrals over the Brillouin zone. Using symmetry this reduces to integrals over the irreducible wedge of the zone. These integrals are calculated using wave functions and eigenvalues at a finite number of special  $\mathbf{k}$ -points in the Brillouin zone (BZ):

$$\int_{BZ} \frac{d\mathbf{k}}{\Omega_{BZ}} \longrightarrow \sum_k w_k , \quad (3.22)$$

where  $w_k$  is the weight of each  $\mathbf{k}$ -point and  $\Omega_{BZ}$  is the volume of the Brillouin zone. The generation of the special  $\mathbf{k}$ -points is done by the Monkhorst-Pack scheme [109], which generates a uniform  $\mathbf{k}$  point grid. The size of the number of  $\mathbf{k}$ -points is a further parameter of the calculation.

## 3.3 Mapping of potential-energy surfaces

Density-functional theory and the FP-LAPW method allow us to calculate the energy of a system for a *particular configuration* of the nuclei with an error  $\leq 0.1 - 0.2$  eV for energy differences. For dynamical studies of dissociation processes with moving particles we need the energy for *many configurations* of the nuclei, i.e. the potential-energy surface as a function of the coordinates of the reactants:

$$\text{PES:} \quad \equiv \quad V(\{R_i^j\}) ,$$

where  $R_i^j$  is the coordinate of nuclei  $i$  of species  $j$  taking part in the reaction. As already discussed, potential-energy surfaces are multidimensional functions. In the calculation of dissociation processes of  $\text{H}_2$  on metal surfaces an important approximation can be applied. Due to the large mass mismatch of the  $\text{H}_2$  molecules and the substrate atoms, the surface rearrangement upon adsorption is negligible. Consequently, the PES is a function of the six degrees of freedom of the molecule alone:

$$\text{H}_2/\text{metal-PES} : \quad \equiv \quad V(X_c, Y_c, Z_c, d, \theta, \phi) ,$$

where  $\mathbf{R}_c = (X_c, Y_c, Z_c)$  is the centre of mass of the hydrogen molecule,  $d$  is the distance between the two hydrogen atoms,  $\theta$  and  $\phi$  are the polar and azimuthal angles of the molecule. This approximation reduces the costs of the calculation, because we are able to keep the surface fixed. There are different approaches possible in order to describe potential-energy surfaces with electronic structure calculations. First, the availability of the forces can be used to follow the moving reactants through configuration space. This approach gives usually only information about a very limited region of configuration space, since *ab initio* total energy calculations are very time consuming and it takes even longer to converge the forces than the energies.

Another approach, which we will follow in this work, is to study the PES on a grid of points. Thus, one repeatedly performs *ab initio* total energy calculations of the system for different configurations of the nuclei sampled on a grid and fine tunes this

grid according to the information gained from the sampling. In this way we are able to obtain information about the PES throughout the whole area of configuration space. A common concept in mapping out the PES for  $H_2$  dissociation on metal surfaces [18, 19, 29, 22, 21, 23, 24, 25] is to keep the surface atoms fixed, to change the location, bond length and orientation of the hydrogen molecule and to compute the electronic structure for each geometry. The calculation of a PES mainly reduces to the study of elbow plots: One chooses a configuration  $X_c, Y_c, \theta, \phi$  and varies the coordinates  $Z_c$  and  $d$ . Another important aspect of *ab initio* total energy calculations of a PES for the dissociation process is to concentrate the study on the dissociation over high-symmetric sites on the surface, i.e. the bridge, hollow and on-top sites as indicated in Fig. 3.4. This ansatz is motivated by the assumption that due to their distinct symmetry these points should represent the major features like minimum and maximum energy barriers of the PES rather well. An important advantage of these sites are the significantly lower costs of the calculations due to their high symmetry. The usual mapping approach of a six-dimensional PES can be summarised as follows:

- (1) Choose high symmetric sites and fix the four coordinates  $X_c, Y_c, \theta, \phi$ .
- (2) Scan the other two dimension  $d$  and  $Z_c$  based on a grid.
- (3) Adjust the grid in order to scan the energy barriers in detail.

An example of the sampling of the energies within one elbow plot can be found in Fig. 3.5. We plotted a two-dimensional cut through the PES for hydrogen dissociation on the potassium covered Pd(100) surface. The region of the energy barrier where the bond of the molecule starts to break is scanned in more detail.

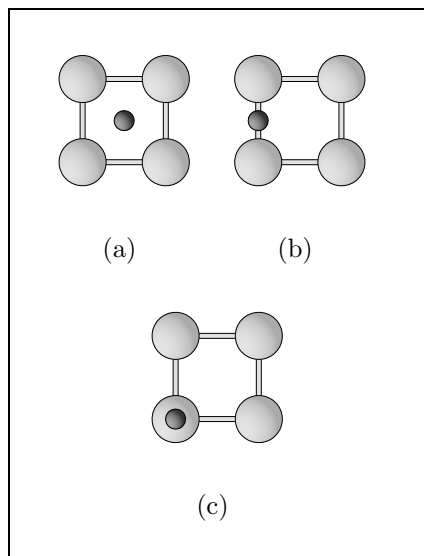


Figure 3.4: Adsorption sites for an atom on the  $(1 \times 1)$  surface. (a): atom above the hollow site. (b) atom above bridge site. (c) atom on top site.

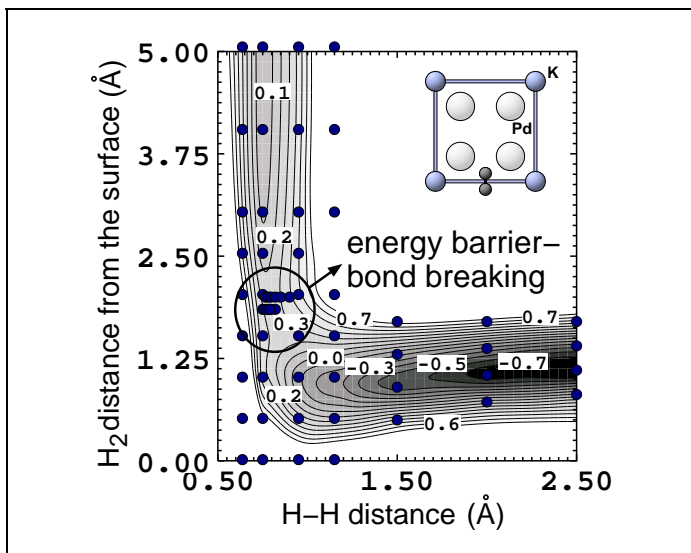


Figure 3.5: Sampling of an elbow plot of the *ab initio* PES for  $H_2$  dissociation on a  $K(2 \times 2)/Pd(100)$  surface. The sampling is based on a grid of points. The region of the energy barrier is scanned in detail. Inset: orientation of the molecule.

---

## Chapter 4

### Interpolation of potential-energy surfaces

---

The potential-energy surface is the central quantity in theoretical studies of chemical processes. Due to the development of efficient algorithms based on density-functional theory and also due to an improvement of computer power it has been possible to map out detailed potential-energy surfaces of the dissociation of hydrogen on metal surfaces in recent years [18, 19, 29, 22, 21, 23, 24, 25]. However, *ab initio* calculations of a PES are accurate, but also very time-consuming and therefore provide us only with a limited and discrete set of potential energies. The number of total energies calculated in theoretical studies ranges typically from 100 to 1000 *ab initio* energies.

On the other hand, in molecular dynamics simulations we have to solve the Schrödinger equation, in the quantum-mechanical case, or Newton's equations of motion, in the classical approximation, for many different configurations of the atoms. The statistical nature of the problem can require to evaluate the energy and the forces of the reactants up to  $10^6 - 10^7$  times, which is not practical for a direct *ab initio* application. In order to bridge this gap the *discrete set* of potential energies needs to be interpolated to a *continuous function* of the configuration of the reactants, i.e. from a discrete *ab initio* PES we want to govern a continuous *ab initio* PES.

Because of the importance of potential-energy surfaces in theoretical studies of reaction dynamics, the interpolation scheme should fulfil a number of requirements:

- The method must be *accurate*, i.e. its error for energy differences should be in the range of density-functional theory implementations based on the actual approximation of the exchange-correlation functional ( $\leq 0.1 - 0.2$  eV).
- The evaluation of the interpolated energy needs to be *fast* to allow immediate molecular dynamics (MD) calculations.
- The computation of the *forces* has to be available to allow MD-calculations.
- The *number of energies* required for accurate fitting should be orders of magnitude lower compared to direct, on-the-fly *ab initio* MD ( $\leq 10^6 - 10^7$  points).
- The fitting procedure itself should be automatic, not manual.
- It should be *flexible* to allow fitting a detailed high-dimensional PES.
- It should conceptually allow its extension to *high dimensions* ( $\geq 6$ ).

- Ideally the method should be *general* to allow its application to a wide range of problems.

## 4.1 Interpolation schemes

Various approaches to fit a PES can be found in the literature. A common feature is the optimisation of a number of parameters to reproduce the underlying data with minimum error.

A London-Eyring-Polanyi-Sato (*LEPS*) form has been used widely in the last decades [42, 43, 44, 45]. It is a two-state binding model describing the interactions by Morse-potentials. Coulomb and exchange integrals are chosen such that the potential describes the intramolecular binding for  $z \rightarrow \infty$  and the atomic chemisorption potential at large interatomic distances  $d$  and small distances from the surface  $z$ . An advantage of this method is its simplicity, which is also a disadvantage, since it is not flexible enough to describe a detailed high-dimensional PES.

If a large amount of data is available, a fit can be made to very general functions, such as *cubic splines* [57]. However, conventional splines are not very robust with respect to noise in the data and both the effort needed to fit the data with a spline function and the interpolation error increase rapidly with increasing dimensions.

A genetic algorithm has been proposed for the fitting of a PES [46]. It is designed to search for the correct functional form of the PES in a tree like structure of simple basis functions. It is a general scheme and the evaluation of the potential after fitting is fast. However, it has been successfully employed only for problems with up to three dimensions. Furthermore a good initial guess of the functional form is needed as a starting point for the search.

Another approach is to assume a functional form with a modest number of adjustable parameters [41, 51]. A well-chosen function may allow effective fits to small amounts of data, but finding such a function requires intuition and a deep insight into the problem.

Recently a hybrid scheme using a combination of numerical and analytical interpolation techniques has been proposed for hydrogen dissociation processes [48, 28, 27]. Instead of parameterising the *ab initio* PES directly, the interpolation is done after subtracting the potential experienced by two isolated H atoms, located at the same positions as the atoms in the H<sub>2</sub> molecule. This results in a smooth and weakly site dependent function, which is easier to interpolate than the strongly corrugated dissociation PES. The obtained effective six-dimensional potential-energy surface is then approximated by a combination of splines and analytical functions. The splines are used to fit the two-dimensional elbow plots, the analytical functions are applied to represent the other four dimensions. Yet, this approach needs insight and it might be difficult to apply it to the interpolation of higher dimensional PESs.

As a concrete example of fitting a potential-energy surfaces we will briefly present the interpolation of the PES of hydrogen on the clean palladium surface, H<sub>2</sub>/Pd(100), calculated by Wilke and Scheffler [29, 22] with analytical functions [41] and a tight-binding scheme [47].



## 4.2 H<sub>2</sub>/Pd(100) Potential-Energy Surface: Interpolation with analytical functions

Fitting a PES with analytical functions requires the choice of some basis functions. In order to choose the appropriate functions, exceptional intuition and insight into the problem is required. An important concept for describing dissociation in the plane of the elbow plots, the  $Zd$  plane, is the introduction of reaction path coordinates [110]. This enables a division of the motion of the molecule into that along the reaction path (coordinate  $s$ ), the solid line in Fig. 2.1, p. 14, and into a vibration perpendicular to this curve (coordinate  $r$ ). At large distances of the molecule from the surface the vibration is that of the gas-phase molecule, when the atoms are chemisorbed on the surface it is that of the products. The potential can then be separated into three parts, one for the corrugation of the potential in lateral directions, one for the rotation and one for the vibration [41]:

$$V(X_c, Y_c, s, r, \theta, \phi) = V^{corr} + V^{rot} + V^{vib} \quad (4.1)$$

with:

$$V^{corr} = \sum_{m,n=0}^2 V_{m,n}^{(1)}(s) \cos mGX_c \cos nGY_c, \quad (4.2)$$

$$\begin{aligned} V^{rot} = & \sum_{m=0}^1 V_m^{(2)}(s) \frac{1}{2} \cos^2 \theta (\cos mGX_c + \cos mGY_c) \\ & + \sum_{n=1}^2 V_n^{(3)}(s) \frac{1}{2} \sin^2 \theta \cos 2\phi (\cos nGX_c - \cos nGY_c) \end{aligned} \quad (4.3)$$

$$V^{vib} = \frac{\mu}{2} \omega^2(s) [r - \Delta r(X_c, Y_c, s)]^2. \quad (4.4)$$

$G = 2\pi/a$  is the basis vector of the reciprocal lattice,  $a$  is the lattice constant,  $\mu$  is the reduced mass of the hydrogen molecule, and  $\omega(s)$  is the frequency perpendicular to the reaction path. The functions  $V_{m,n}^{(i)}(s)$ ,  $\omega(s)$ , and  $\Delta r(X_c, Y_c, s)$  are the 'parameters' of the calculation. They are determined such that the difference between the analytical potential and the 250 *ab initio* total energies is on the average smaller than 25 meV, which is within the accuracy of the underlying *ab initio* data. The sine and cosine terms represent the symmetry of the surface.

We will not discuss the single parts of the potential in further detail, but rather point out that the choice of the appropriate basis functions and also of the functions  $V_{m,n}^{(i)}(s)$ ,  $\omega(s)$ , and  $\Delta r(X_c, Y_c, s)$  is a formidable task and can be even more cumbersome for higher dimensions. Furthermore, the fit is tailored to the specific application and is not a general scheme. If, e.g. the motion of the surface atoms plays a role in the dissociation process, this parameterisation has to be extended.

### 4.3 H<sub>2</sub>/Pd(100) Potential-Energy Surface: Interpolation with a tight-binding scheme

Recently the PES for the system H<sub>2</sub>/Pd(100) has been fitted with a non-orthogonal tight-binding total energy method (TBTE) [47]. We will briefly introduce the basics of the TBTE method. For a more detailed description please refer to the publications by Cohen, Mehl and Papaconstantopoulos [111, 112, 113].

According to density functional theory, using the Kohn-Sham *ansatz* for the kinetic energy, the total energy of a system of electrons moving in a solid can be written as:

$$E[n(\mathbf{r})] = \sum_i f(\mu - \varepsilon_i) \varepsilon_i + F[n(\mathbf{r})], \quad (4.5)$$

where  $n(\mathbf{r})$  is the electron density,  $\varepsilon_i$  is the Kohn-Sham eigenvalue of the  $i^{\text{th}}$  electronic state,  $\mu$  is the chemical potential and the sum is over all electronic states of the system. The function  $f$  is a Fermi-function. The functional  $F[n(\mathbf{r})]$  contains the ion-ion interaction energy, the parts of the Hartree and Exchange-Correlation energy not included in the eigenvalue sums, and corrections for double counting in the eigenvalue sums. Commonly the eigenvalue sum is calculated by tight-binding methods, and  $F[n(\mathbf{r})]$  is approximated by a sum of pair potentials [112]. The TBTE scheme does not include a pair-potential term. Instead, the method takes advantage of the fact that the total energy of a system is independent of the choice of zero of the potential. In the TBTE method the total energy becomes [112]:

$$E[n(\mathbf{r})] = \sum_i f(\mu' - \varepsilon'_i) \varepsilon'_i, \quad (4.6)$$

where  $\mu'$  is the shifted chemical potential and  $\varepsilon'_i$  are the shifted eigenvalues of a generalised Schrödinger equation:

$$(\hat{\mathbf{H}} - \varepsilon'_i \hat{\mathbf{S}}) \Psi_i = 0, \quad (4.7)$$

where  $\hat{\mathbf{H}}$  and  $\hat{\mathbf{S}}$  are the Hamiltonian and overlap matrices in an atomic basis representation  $\{\Phi_\alpha\}$ . The Hamiltonian and overlap matrices are assumed to have the Slater-Koster form [114], which have a physical meaning. The Slater-Koster integrals are the parameters of this fitting scheme.

An important advantage of the TBTE method is that the quantum-mechanical nature of bonding is taken into account. Therefore a good fit for the system H<sub>2</sub>/Pd(100) has been obtained by using 55 *ab initio* points only. But since the diagonalisation of matrices is required the computational effort is two to three orders of magnitude higher compared to analytical parameterisations.

### 4.4 Advantages and disadvantages of various interpolation schemes

Interpolation of potential-energy surfaces is the key problem for theoretical discussions of reaction dynamics. All interpolation schemes have some advantages and

disadvantages. In order to give a comprehensive overview over the different methods, we recapitulated the characteristics of the discussed fitting procedures in table 4.1. The major disadvantages are that the different methods are either not very flexible, like the LEPS potential, they are not general like fits with analytical functions, their evaluation is very time-consuming, like the tight-binding scheme or their error increases rapidly with increasing dimensions, like cubic splines. It is therefore fair to say that the issue of finding the optimal interpolation tool despite its importance is not yet solved.

	Advantages	Disadvantages
LEPS potential	<ul style="list-style-type: none"> <li>○ simple form</li> </ul>	<ul style="list-style-type: none"> <li>○ not flexible enough for high dimensional problems</li> </ul>
Splines	<ul style="list-style-type: none"> <li>○ general</li> <li>○ very flexible</li> <li>○ allow fast evaluation of the energy after fitting</li> </ul>	<ul style="list-style-type: none"> <li>○ require large amount of data</li> <li>○ not very robust against noise</li> <li>○ error increases rapidly with increasing dimension</li> </ul>
Genetic algorithm	<ul style="list-style-type: none"> <li>○ general (in principle, but not yet shown)</li> <li>○ evaluation of the potential fast</li> </ul>	<ul style="list-style-type: none"> <li>○ good guess of the functional form needed</li> <li>○ only applied to problems with up to three dimensions</li> </ul>
Analytical Functions	<ul style="list-style-type: none"> <li>○ modest number of parameters</li> <li>○ suitable for calculations of the potential energy in extensive statistics</li> <li>○ fast to evaluate after fit</li> </ul>	<ul style="list-style-type: none"> <li>○ often not flexible enough to fit detailed PES</li> <li>○ choice of set of basis functions in high dimensions difficult</li> <li>○ not general</li> </ul>
Tight-binding scheme	<ul style="list-style-type: none"> <li>○ treatment of the kinetic and covalent interactions of electrons chemically more appropriate</li> <li>○ low number of points needed for fit</li> </ul>	<ul style="list-style-type: none"> <li>○ requires the diagonalisation of Hamilton and overlap matrices <math>\Rightarrow</math> slower than other schemes</li> <li>○ fitting procedure highly-nonlinear and difficult</li> </ul>

Table 4.1: Comparison of different interpolation tools for potential-energy surfaces: advantages and disadvantages

---

## Chapter 5

### Neural Networks

---

Despite its importance a general, fast and accurate interpolation tool for potential-energy surfaces is still lacking. As an alternative to the fitting schemes discussed in section 4.1, we will now introduce an interpolation method based on multilayer, feed-forward neural networks (NN) [52,53,54]. Neural Networks can be described as general, non-linear fitting functions that do not require any assumptions about the functional form of the underlying problem. The resulting interpolant is smooth, continuous and leads to a relatively simple but accurate expression for the PES. Evaluation of the potential with a fitted neural network is cheap and the derivatives of the potential surface are easily obtainable. Therefore, if the number of parameters and required data for a good fit scale favourably with dimension, neural networks will be ideal for molecular dynamics applications.

Artificial neural networks have been used for many years in physics and chemistry in a variety of applications. They range from the classification of stars and galaxies, the classification of mass spectra of organic compounds, the classification of events in high-energy physics, protein-structure prediction, analysis of DNA sequences, design of pharmaceuticals, robotics, to the coupling in reaction-diffusion systems and many more [12,13,14]. Applications in other fields include the prediction of time-series in finance or geophysics [115,116], signal processing and process control in engineering, thinking, consciousness and memory in neuro-physiology and character or speech recognition for various tasks. Artificial neural networks are also used in a number of commercial applications.

The main area of research in neural computing is devoted to classification or pattern recognition problems which is a profoundly different task from the one we are interested in, the interpolation of a multidimensional function. In the former, where neural networks have proven to be extremely useful, the network has to decide whether something is true or not true, whereas in the latter it has to map a continuous problem. However, neural networks have also been applied to problems involving function approximation in general [117,118] and more recently to the interpolation of potential-energy surfaces [57,58,59,60,61]. These works concentrate on studies of the potential-energy surface of molecules in the gas phase. We want to extend these applications of the neural network approach to reactions of molecules on surfaces on a high-dimensional PES and employ the neural network PES in molecular dynamics simulations.

It is important to notice that there is no such thing like “the neural network”. Neural networks are rather a class of algorithms inspired from neuro-science with different architectures of the nets. They can also be implemented as hardware [119]. We will focus our work on *multilayer feed-forward neural networks*, which have proven to be useful for function interpolation, see e.g. [57]. In feed-forward nets the information is transmitted through a hierarchical structure of artificial neurons in only one direction. More general architectures exist. Information can be allowed to travel in both ways through the net. These nets are usually called *recurrent networks* and are mainly used to recognise or reproduce time-series.

The fitting procedure to some specific data set is associated with the optimisation of the parameters of the network, which is called *learning* in the neural network context. Learning can proceed in two different ways, *supervised* and *unsupervised*. In supervised learning the optimisation is done on the basis of direct comparison of the output of the network with known correct answers. In unsupervised learning the learning goal is not defined in terms of specific correct examples but rather through correlations of the input data, which is more difficult. Since for fitting a PES a specific data set for learning, the energies from the *ab initio* total energy calculations, is available this chapter will be concerned with supervised learning in feed-forward neural networks.

## 5.1 Inspiration from Neuro-science

As the term “neural network” implies, this approach was aimed towards modelling real networks of neurons in the brain and were inspired by a number of features of the brain that would be desirable in artificial systems, like its robustness and fault tolerance, its flexibility, its ability to deal with fuzzy and noisy information and its highly parallel structure.

The brain is composed of about  $10^{11}$  neurons. Figure 5.1(a) is a schematic drawing of a single biological neuron. Networks of nerve fibre called *dendrites* are connected to the *cell-body*. Extending from the cell body is a single long fibre called *axon*. They end into the *synaptic junctions*, or *synapses*, to other neurons. Transmission of a signal from one cell to another is a complex chemical process. The effect is to raise or lower the electrical potential inside the body of the receiving cell. If this potential reaches a threshold, a pulse is sent down the axon, the neuron *fires*. *Learning* means in this context a strengthening or weakening of the synaptic junctions between the neurons.

McCulloch and Pitts [120] proposed a simple model of a neuron as a binary threshold unit, see Fig. 5.1(b). The model neuron computes a weighted sum of the inputs from other units, and either computes a one or a zero according to whether the sum is above or below a certain threshold  $\mu_j$ :

$$n_j(t+1) = \Theta \left( \sum_i w_{ij} n_i(t) - \mu_j \right), \quad (5.1)$$

where  $n_j(t)$  is the state of neuron  $j$  and time  $t$  and  $\Theta(x)$  is the unit step function or Heaviside function. The weight  $w_{ij}$  represents the strength of the synapse connecting neuron  $i$  to neuron  $j$ . Of course, real neurons involve many complications omitted in

the McCulloch-Pitts description. The model is only *inspired* by the brain but not a one-to-one description. A network composed of a number of such neurons is able to describe linear-separable problems only. A simple generalisation of the McCulloch-Pitts equation (5.1) is the use of a continuous-valued non-linear function  $f(x)$ , the so-called *activation* or *transfer function*:

$$n_j = f \left( \sum_i w_{ij} n_i - \mu_j \right) . \quad (5.2)$$

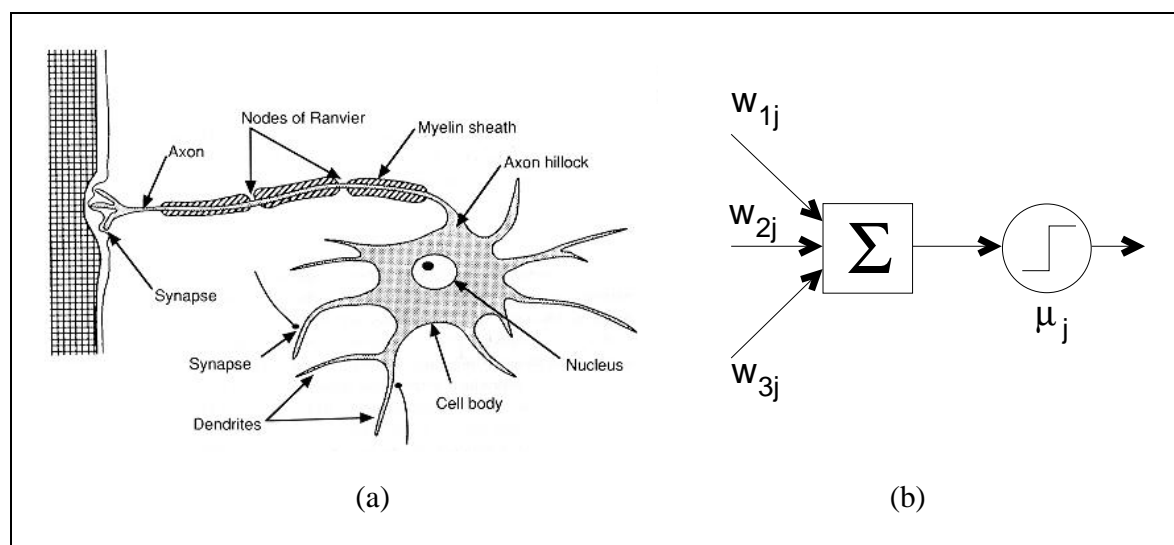


Figure 5.1: Schematic drawing of a biological and an artificial neuron. (a) Schematic view of a biological neuron from the brain. (b) Schematic diagram of a McCulloch-Pitts neuron. The unit fires if the weighted sum  $\sum_i w_{ij} n_i$  of the inputs reaches or exceeds the threshold  $\mu_j$ .

## 5.2 Structure of feed-forward neural networks

An artificial neural network is an example of a highly flexible non-linear model. It consists of a number of artificial neurons or *nodes*, typically arranged in layers, interconnected via a set of links. A schematic representation of such a net is plotted in Fig. 5.2. Each link multiplies its input by a parameter, the weight, before supplying it to a new node. Each node sums over its inputs and applies a function to the resulting value. In the *input layer* the identity function is used to distribute the information to the second layer. This layer is called the *hidden layer* because its input and output is not visible from to the outside world. The hidden layer is the core of the non-linear fitting of the data set. The *output layer* collects the information from the hidden layer and transforms it again. This network design, in which every node is connected to every node in the adjacent layers but nodes in the same layer are not connected and the information is transmitted only in one direction, is called a *multilayer feed-forward neural network*.

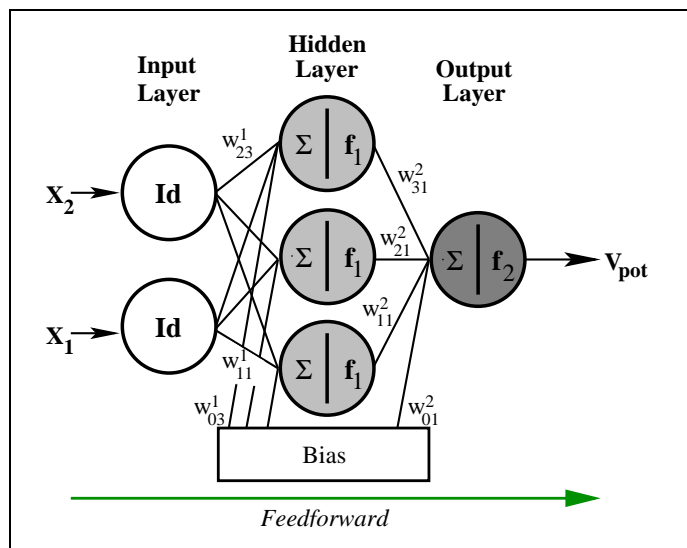


Figure 5.2: Schematic architecture of an artificial neural network, here a multilayer feedforward net. The neurons are arranged in layers. The bias unit acts as an adjustable offset. The output function of this network with non-linear basis functions  $f_{1,2}(x)$  is:  $V_{pot}(x_1, x_2) = f_2 \left( w_{01}^2 + \sum_{j=1}^3 w_{j1}^2 f_1 \left( w_{0j}^1 + \sum_{i=1}^2 w_{ij}^1 x_i \right) \right)$ .

The output  $y_j^p$  of the  $j$ th neuron in layer  $p$ , which receives inputs  $\{y_i^{p-1}\}$  via links with weights  $\{w_{ij}^p\}$  from the previous layer, may be written as:

$$y_j^p = f \left( x_j^p \right) = \left( w_{0j}^p + \sum_i w_{ij}^p y_i^{p-1} \right), \quad (5.3)$$

where  $f()$  is the non-linear activation function. The weight  $w_{0j}^p$  is called the *bias* of node  $j$  in layer  $p$ , which acts as an adjustable offset of the activation function. The input layer is counted as layer 0, the output layer corresponds to layer  $N$ . The output of a fully connected three layer neural network with one input, one hidden and one output layer and  $n_0, n_1$  and  $n_2$  nodes in each layer, respectively, can be written as:

$$y_k^2 = f_2 \left( w_{0k}^2 + \sum_{j=1}^{n_1} w_{jk}^2 f_1 \left( w_{0j}^1 + \sum_{i=1}^{n_0} w_{ij}^1 y_i^0 \right) \right) \quad \forall k = 1, 2, \dots, n_2. \quad (5.4)$$

In the case of fitting a potential-energy surface with a neural network, the  $\{y_i^0\}$  represent the coordinates of the reactants. For hydrogen dissociation the input layer of the net will consist of six units corresponding to  $\{y_i^0\} = \{X_c, Y_c, Z_c, r, \theta, \phi\}$ . In the output layer we will have just one output node, the potential energy  $V_{pot}(\{y_i^0\})$ .

For convenience we have presented in Fig. 5.2 a feed-forward network with only one hidden layer. However, it should be noted that the number of hidden layers is not restricted. In fact, we will most often use networks with two hidden-layers.

A generalisation of the network architecture is to employ direct links from the input to the output units in conjunction with a linear activation function in the outputs. This can be especially helpful to speed up learning if some linear dependence is present in the data set.



## 5.3 Function approximation and neural networks

In the approximation of functions with neural networks it is *a priori* not known how many hidden layers and units the net needs to be composed of. Cybenko showed that in order to approximate a particular set of functions with arbitrary accuracy at most two hidden layers with sigmoidal activation functions and a sufficient number of nodes are necessary [121]. It has also been proven, that only one hidden layer is enough to approximate any continuous function to given accuracy [55,56]. The results depend, of course, on how many hidden nodes have to be applied. This is not known in general and needs to be tested for the particular problem.

## 5.4 Activation functions

Non-linear activation or basis functions are what give neural networks their non-linear capabilities. The function must be differentiable for the optimisation of the parameters and we normally want it to saturate at both extremes. The most common forms of activation functions are the monotonically increasing sigmoidal or Fermi-like functions, like the sigmoid or the hyperbolic tangent function:

$$\text{hyperbolic tangent} \quad f(x) = \tanh(x), \quad (5.5)$$

$$\text{sigmoid function} \quad f(x) = \frac{1}{1 + e^{-x}}. \quad (5.6)$$

These functions are also called squashing functions, because of their asymptotic behaviour at  $\pm \infty$ , see Fig. 5.3 and Fig. 5.4. Such activation functions can be employed

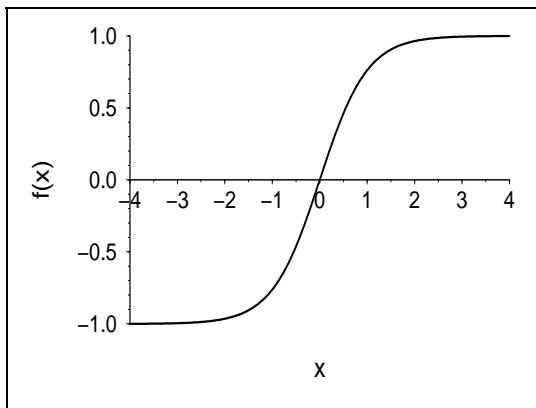


Figure 5.3: *Hyperbolic tangent* with  $f(x) = \tanh(x)$  ranging from  $[-1, +1]$  for  $x \rightarrow \pm \infty$ .

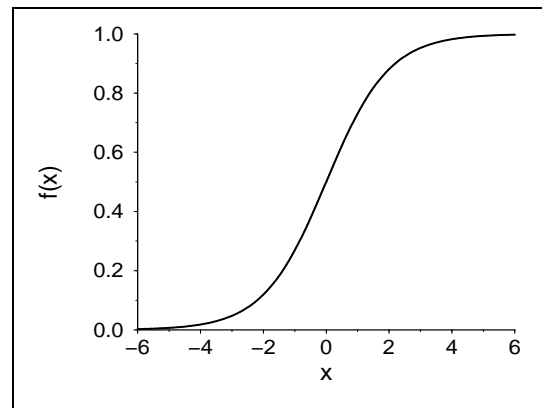


Figure 5.4: *Sigmoid function* with  $f(x) = \frac{1}{1+e^{-x}}$  ranging from  $[0, 1]$  for  $x \rightarrow \pm \infty$ .

in the hidden layers as well as in the output layer. However, one often doesn't want to squash the output values to a certain range and therefore uses a linear output function as the activation function in the output nodes:

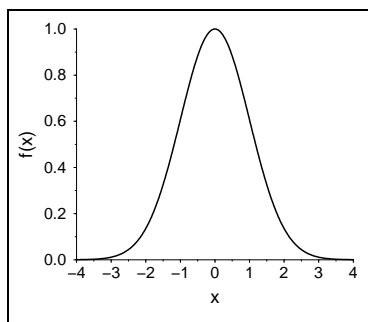
$$\text{linear function:} \quad f(x) = x. \quad (5.7)$$

There is a wide-spread use of both sigmoid and hyperbolic tangent functions in applications published in the literature. However, convergence with functions which are symmetric about the origin, like the hyperbolic tangent, is often faster [54] and therefore will be preferred (for a more detailed discussion see Sec. 5.9, p. 56).

In order to describe the network architecture in a simple way the following notation is used: the number of nodes in the layers, followed by letters denoting the activation function, with  $s$  for sigmoid,  $l$  for linear and  $t$  for the hyperbolic tangent. In this notation, the network in Fig. 5.2 in conjunction with a hyperbolic tangent function in the hidden layer and a linear function in the output layer has a  $\{2-3-1\}tl$  structure.

Although most systems use nodes based on dot products and sigmoidals, many other types of nodes like the *radial basis function* (RBF) network can be used [122, 123]. In RBF networks, the dot product of the weight and the input vector ( $\sum_{i=1}^p w_{ij}^p y_i^{p-1}$ ) is replaced by an Euclidean distance between the input and the weight ( $\|y_i^{p-1} - w_{ij}^p\|_2$ ). Furthermore the sigmoidal function is replaced by a Gaussian. The output  $y_j^p$  of the  $j$ -th neuron in layer  $p$  is computed as:

$$y_j^p = \exp\left(-\frac{\|y_i^{p-1} - w_{ij}^p\|_2}{2(\sigma_j^p)^2}\right). \quad (5.8)$$



**Figure 5.5:** *Gaussian*

The Gaussian function is plotted in Fig. 5.5. Radial basis function networks form a separate class of nets. The parameters have a different meaning in comparison to the sigmoidal networks. The weight  $w_{ij}^1$  can now be interpreted as the mean of the  $j$ -th Gaussian and  $\sigma_j$  is the corresponding standard deviation. They have to be determined by unsupervised clustering algorithms, which make their application more difficult. Unlike sigmoidal nodes which can cover the entire space, a single RBF unit covers only a small region of the input space. This can be an advantage, because learning can be faster, although this is highly problem dependent.

However, the locality property of RBFs may be a disadvantage particularly in high dimensional spaces because many units are needed to cover the spaces and the output will be rather *bumpy*. Therefore RBFs are more appropriate for low dimensional problems, for fitting high-dimensional PESs we will prefer sigmoidal activation functions.

## 5.5 Supervised Learning - parameter optimisation

The training of a feed-forward neural network is equivalent to performing a non-linear optimisation of the network parameters, the weights. Due to the inspiration from neuroscience, this process is called learning in the network community. A special form of parameter optimisation is the so-called *supervised learning*. The learning is done by comparing the output with known correct answers. In the case of fitting a PES, the known answers are the energies obtained from *ab initio* total energy calculations. The optimisation of the network weights is performed by some iterative optimisation scheme

until a desirable solution measured by a scalar cost function  $E$  is reached. The cost function is normally taken as the sum of the squared residuals between the true or targeted value  $\mathbf{t}(\{y_i^0\})$  and the actual output of the network  $\mathbf{y}(\{w_{ij}\}, \{y_i^0\})$  depending on the inputs and the weights:

$$E(\{y_i^0\}, \{w_{ij}\}) = \sum_{j=1}^n \|\mathbf{t}(\{y_i^0\}) - \mathbf{y}(\{w_{ij}\}, \{y_i^0\})\|_2, \quad (5.9)$$

where  $n$  is the number of examples in the data set. In order to minimise the costs the network cycles repeatedly through the following steps of the learning process:

- (1) present the network one example of the set of data,
- (2) measure the response of the output layer of the net,
- (3) calculate the mean squared error between the output and the target value,
- (4) adjust the weights to minimise the cost function
- (5) if the mean squared error reaches a desired lower bound, stop the iteration, otherwise go to (1).

The optimisation of the weights is usually done by gradient-based learning methods like steepest decent, conjugate gradients or quasi-newton algorithms [124]. Two different update schemes of the parameters in (4) exist. One can first present the network the whole set of examples, called an *epoch*, and only then changes the weights accordingly, known as *batch* or *off-line* learning, or the update is performed after the presentation of every single example, chosen randomly. This is called *stochastic* or *on-line* learning. On-line and off-line learning are compared in table 5.1. There are several advantages of stochastic over batch learning. It is often faster because redundancy or the presence of

On-line	Off-line
usually much faster than batch or off-line learning	conditions of convergence well understood
results obtained often better in comparison to off-line learning	many optimisation schemes like conjugate gradients only operate in off-line mode
importance of each data point directly visible	theoretical analysis of convergence rates simpler

Table 5.1: *On-line and off-line learning for feed-forward neural networks*

patterns which are very similar lead to slow convergence in off-line learning. Furthermore, it results most likely in better solutions, because updating the weights after each example increases the probability of getting out of a local minimum of the error surface before the iteration gets stuck. Off-line learning will discover the minimum wherever

the weights are. In on-line learning, the noise present in the updates coming from the numerical calculation of the gradient after each example can result in weights which are able to find a different minimum. However, this noise, which is so crucial for finding better local minima also prevents full convergence to the minimum.

## 5.6 Optimisation algorithms

When each training example is presented to the network, an output value depending on the current set of weights is predicted and the error to the targeted value is calculated. The training algorithms adjust the weights to reduce the output error as measured by the cost function. Given the structure of the network, it is relatively easy to determine the derivative of the output with respect to the weights. These derivatives can be used to minimise the output error of the network.

### 5.6.1 Basics of gradient descent

Let us start with a one-dimensional case, where the update equation for gradient descent can be written as:

$$w(t) = w(t-1) - \eta \frac{\partial E}{\partial w}, \quad (5.10)$$

where  $E$  is the error function of the network,  $w$  is the parameter vector and  $\eta$  is the learning rate. In order to find out something about the optimal learning rate  $\eta_{opt}$  we expand the error function  $E$  in a Taylor series about the current weight  $w_c$ :

$$E(w) = E(w_c) + (w - w_c) \frac{dE(w_c)}{dw} + \frac{1}{2} (w - w_c)^2 \frac{d^2E(w_c)}{d^2w} + \dots, \quad (5.11)$$

with  $\frac{dE(w_c)}{dw} = \frac{dE}{dw}|_{w=w_c}$ . If  $E$  is quadratic the second order derivative is constant and the higher order terms vanish. Differentiating both sides with respect to  $w$  then gives:

$$\frac{dE(w)}{dw} = \frac{dE(w_c)}{dw} + (w - w_c) \frac{d^2E(w_c)}{d^2w}. \quad (5.12)$$

Setting  $w = w_{min}$  and noting that  $dE(w_{min})/dw = 0$  gives after some rearrangement:

$$w_{min} = w_c - \left( \frac{d^2E(w_c)}{d^2w} \right)^{-1} \frac{dE(w_c)}{dw}. \quad (5.13)$$

If we compare this with the gradient update equation (5.10), we find:

$$\eta_{opt} = \left( \frac{d^2E(w_c)}{d^2w} \right)^{-1} \quad (5.14)$$

If  $E$  is not exactly quadratic then the higher order terms in equation (5.11) do not vanish and (5.13) is only an approximation. However, convergence can still be quite fast. The right side of equation (5.14) is the inverse of the Hessian matrix  $H^{-1}$ , with:

$$H_{ij} = \frac{\partial^2 E}{\partial w_i \partial w_j}. \quad (5.15)$$

The eigenvectors of  $H$  point in the directions of the major and minor axes, the eigenvalues measure the steepness of  $E$  along the corresponding eigendirection. In multiple dimensions it is rather time consuming to calculate the Hessian, therefore the different minimisation algorithms based on gradient-descent vary by the approximations made to estimate the learning rate  $\eta$ .

### 5.6.2 Backpropagation

The simplest minimisation procedure in gradient descent is to take the learning rate  $\eta > 0$  as a scalar constant:

$$\mathbf{w}(t) = \mathbf{w}(t-1) - \eta \frac{\partial E}{\partial \mathbf{w}} . \quad (5.16)$$

This minimisation is called *backpropagation* in the neural network community. It is essentially the standard steepest-descent algorithm [124]. More sophisticated procedures use variable  $\eta$ . A momentum term can increase the speed of convergence when the cost surface is highly non-spherical because it damps the size of the steps along directions of high curvature. Therefore it yields a larger learning rate along the directions of low curvature. Momentum is a method of averaging the weight changes. We first calculate the change that would correspond to the direction of steepest descent at the current weight. But the change we actually make is the average between this indicated change and the last actual change. We accomplish this by using an exponential average:

$$\Delta \mathbf{w}(t) = (1 - \mu) \eta \frac{\partial E_t}{\partial \mathbf{w}} - \mu \Delta \mathbf{w}(t-1), \quad (5.17)$$

For the update of the weights in (5.16) and (5.17) the partial derivative of the error function with respect to the weights needs to be calculated. We will now sketch how this can be done and why this gradient-descent algorithm is called *backpropagation*.

The cost function of the neural network is the sum over the squared residuals of the target value  $t_k$  and the current value  $y_k^{N=2}$  of the  $k$ -th output node for each example  $p$ :

$$E = \sum_p E_p(\mathbf{w}, \mathbf{y}^0) , \quad (5.18)$$

with:

$$E_p = \frac{1}{2} \sum_k (t_k - y_k^N(\mathbf{w}, \mathbf{y}^0))^2 . \quad (5.19)$$

This cost function needs to be minimised with respect to the weights, i.e.:

$$\frac{\partial E}{\partial \mathbf{w}} = \sum_p \frac{\partial E_p(\mathbf{w}, \mathbf{y}^0)}{\partial \mathbf{w}} = 0 . \quad (5.20)$$

Therefore we need to calculate the derivatives  $\frac{\partial E_p}{\partial \mathbf{w}}$ . In order to derive this derivative, let us assume a three-layer network as plotted in Fig. 5.2, p. 44. In the network, the

following relationships hold for the first and the second layer respectively, where the input layer is counted as the 0-th layer:

$$\text{Input - hidden layer} \quad \begin{cases} x_j^1(\mathbf{w}, \mathbf{y}^0) = \sum_i^{n_0} w_{ij}^1 y_i^0 \\ y_j^1(\mathbf{w}, \mathbf{y}^0) = f_j^1(x_j^1) \end{cases}, \quad (5.21)$$

$$\text{Hidden - output layer} \quad \begin{cases} x_k^2(\mathbf{w}, \mathbf{y}^0) = \sum_j^{n_1} w_{jk}^2 y_j^1 \\ y_k^2(\mathbf{w}, \mathbf{y}^0) = f_k^2(x_k^2) \end{cases}, \quad (5.22)$$

where we dropped the bias terms for simplicity. The output of each neuron depends on the network input vector  $\mathbf{y}^0$  and the current weight vector  $\mathbf{w}$ . The partial derivative of the error function (5.19) with respect to a weight leading into an output node can be calculated using the chain rule:

$$\frac{\partial E_p}{\partial w_{jk}^2} = (t_k - y_k^2(\mathbf{w}, \mathbf{y}^0)) \frac{\partial y_k^2}{\partial w_{jk}^2} = (t_k - y_k^2(\mathbf{w}, \mathbf{y}^0)) \frac{\partial f_k^2}{\partial x_k^2} \frac{\partial x_k^2}{\partial w_{jk}^2}. \quad (5.23)$$

Consequently, the partial derivative of the error function with respect to a weight leading into a hidden node is (with  $n_2$  output nodes):

$$\frac{\partial E_p}{\partial w_{ij}^1} = \sum_{k=1}^{n_2} (t_k - y_k^2(\mathbf{w}, \mathbf{y}^0)) \frac{\partial f_k^2}{\partial x_k^2} \frac{\partial x_k^2}{\partial y_j^1} \frac{\partial f_j^1}{\partial x_j^1} \frac{\partial x_j^1}{\partial w_{ij}^1}. \quad (5.24)$$

In order to calculate the error derivative, the error is propagated back through the network and on the way the partial derivatives are being collected. This is why this gradient-descent based minimisation scheme is called *backpropagation* in the neural network community. For a more general derivation of the backpropagation update formulas please refer to Appendix A, p. 149.

### 5.6.3 Conjugate Gradients

Conjugate gradient is a standard first order optimisation scheme, see [124], i.e. it uses information on the first derivative of the error function only. Its main properties are:

- it is an  $O(n)$  method,
- it attempts to find descent directions that try to minimally spoil the result achieved in the previous iterations,
- it uses a line search, a one-dimensional sub-minimisation,
- it works only for batch learning .

It has been applied with success in multi-layer network training on problems that are moderately sized with rather low redundancy in the data. However, its main disadvantage remains that it is an off-line method only.

### 5.6.4 Newton and Quasi-Newton methods

We have seen that the weight update can be computed in the following way:

$$\Delta \mathbf{w} = \eta \left( \frac{dE^2(\mathbf{w})}{d^2\mathbf{w}} \right)^{-1} \frac{dE(\mathbf{w})}{d\mathbf{w}} = \eta (\mathbf{H}(\mathbf{w}))^{-1} \mathbf{J}(\mathbf{w}), \quad (5.25)$$

where  $\mathbf{J}$  is the Jacobi matrix,  $\mathbf{H}$  is the Hessian matrix and  $\eta$  must be chosen in the range of  $0 < \eta < 1$ , since  $E$  is in practice not perfectly quadratic. The Newton algorithm uses the Hessian matrix explicitly. However, one of the main drawbacks is that an  $N \times N$  Hessian matrix, where  $N$  is the number of weights, must be stored and inverted, which takes  $O(N^3)$  iterations. Furthermore, there is no guarantee of convergence. Since the error function is in general not quadratic, the Hessian might not be always positive definite and the algorithm diverges. In Quasi-Newton algorithms, the inverse Hessian is estimated by some positive definite matrix, which ensures convergence. In the Gauss-Newton algorithm the Hessian is approximated by the square Jacobian matrix [124, 54]:

$$\hat{\mathbf{H}}(\mathbf{w}) = \mathbf{J}(\mathbf{w}) \mathbf{J}^T(\mathbf{w}), \quad (5.26)$$

and the update is therefore:

$$\Delta \mathbf{w} = \eta \left( \mathbf{J}(\mathbf{w}) \mathbf{J}^T(\mathbf{w}) \right)^{-1} \mathbf{J}(\mathbf{w}). \quad (5.27)$$

Optimisation schemes which approximate the Hessian of the system are called second order algorithms, since the Hessian is the matrix of the second derivatives. Most importantly, Quasi-Newton algorithms are designed for off-line learning. The update step is taken at the end of an *epoch*, i.e. after presentation of the whole data set.

### 5.6.5 Extended Kalman filter

The Kalman filter is a digital filter, an algorithm expressed in state space notation which comes from optimal estimation and control theory [125]. It is designed to filter out noise from time series, e.g. sequences of measurements. However, it has also been applied recently for the optimisation of the weights in neural networks [126, 127, 128]. The Kalman filter exhibits optimal statistical properties when applied to linear dynamic systems containing white noise. It recursively computes minimum variable estimates of the state variables, i.e. the weights. However, feedforward neural network output functions are nonlinear and therefore represent a nonlinear system. The extended version of the Kalman filter models such systems by a local linearisation of the system model around the current estimates of the state variable. Feedforward neural networks can be regarded as a static, non-linear dynamic system, where the optimal weight vector  $\mathbf{w}_{opt}$  is assumed to be invariant with time. For an application of the extended Kalman filter (EKF) to the optimisation of neural networks, we view the presentation of input and output data, and the associated weight estimates, as an evolving time series. In the Kalman filter approach the target output value at presentation  $k$ ,  $\mathbf{t}(\mathbf{x}(k))$ , is assumed to be equal to the sum of the output from the trained network  $\mathbf{y}(\mathbf{w}_{opt}, \mathbf{x}(k))$ , depending on the weight and the input vector, and the vector of modelling errors,  $\mathbf{e}(k)$ :

$$\mathbf{t}(k) = \mathbf{y}(\mathbf{w}_{opt}, \mathbf{x}(k)) + \mathbf{e}(k). \quad (5.28)$$

The extended Kalman filter equations are derived by expanding the nonlinear output function  $\mathbf{y}$  around the current estimate parameter vector  $\hat{\mathbf{w}}(k-1)$ :

$$\mathbf{t}(k) = \mathbf{y}(\hat{\mathbf{w}}(k-1), \mathbf{x}(k)) + \mathbf{J}^T(k) (\mathbf{w}_{opt} - \hat{\mathbf{w}}(k-1)) + \rho(k) + \mathbf{e}(k), \quad (5.29)$$

where  $\mathbf{J}(k)$  is the Jacobi matrix given by:

$$\mathbf{J}(k) = \left. \frac{\partial \mathbf{y}(\mathbf{w}, \mathbf{x}(k))}{\partial \mathbf{w}} \right|_{\mathbf{w}=\hat{\mathbf{w}}(k-1)}, \quad (5.30)$$

and  $\rho(k)$  is the residual of the Taylor expansion of the network output function  $\mathbf{y}$ . The estimate  $\hat{\mathbf{w}}(k)$  is obtained by the optimal regression of  $\mathbf{w}_{opt}$ , i.e. we minimise:

$$\epsilon(k) = \sum_{p=1}^k \|\mathbf{e}(p)\|_2 \lambda^{k-p}, \quad (5.31)$$

where an exponential weighting of the sequence of filter errors is produced by the factor  $\lambda^{k-p}$ . The cost function is determined by summing over all previous iterations, but the weighting factor provides significant weighting only for more recently encountered examples [127]. The time varying forgetting schedule changes the value of  $\lambda$  before the presentation of each training sample according to:

$$\lambda(k) = \lambda_0 \lambda(k-1) + 1 - \lambda_0, \quad (5.32)$$

where  $\lambda_0$  is typically a constant between 0.99 and 0.9995 and the initial value of  $\lambda$ ,  $\lambda(0)$ , is chosen between 0.95 and 0.99. The idea behind the use of a weighting term is

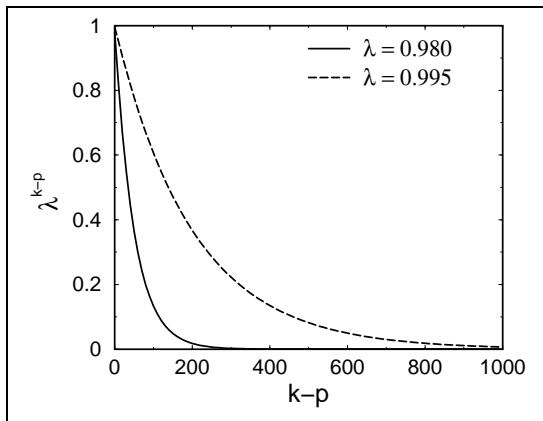


Figure 5.6: Forgetting factor  $\lambda^{k-p}$  for two different values,  $\lambda=0.980$  and  $\lambda=0.995$

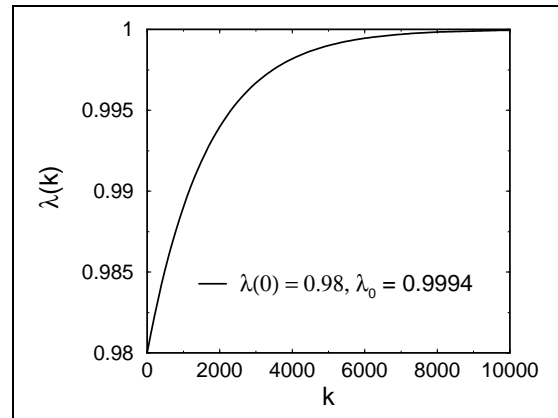


Figure 5.7: Forgetting schedule  $\lambda(k)$ , with  $\lambda(k) = \lambda_0 \lambda(k-1) + 1 - \lambda_0$ , for  $\lambda(0)=0.98$  and  $\lambda_0=0.9994$

twofold. First, one wants to make any contribution from previous iterations negligible at the beginning of the minimisation in order not to get trapped in a local minimum too early during the process. The forgetting factor  $\lambda^{k-p}$  has been plotted in Fig. 5.6 for two different values. At the beginning of the optimisation  $\lambda$  is smaller and fewer examples are taken into account, because  $\lambda$  drops faster to zero. As the number of



epochs increases the weight estimates approach the region of a minimum that corresponds to an acceptable level of error. But then it is advantageous to make use of all available information in the optimisation. Due to the special form of the forgetting schedule  $\lambda(k)$ , more and more examples are included in the cost function as training proceeds. As the number of iterations increases,  $\lambda$  also increases, see Fig. 5.7, and this leads correspondingly to a less pronounced drop in the forgetting factor in Fig. 5.6.

The extended Kalman filter equations are derived from the minimisation of equation (5.31), i.e. by setting

$$\nabla_{\mathbf{w}_{opt}} \epsilon(n) = 0 . \quad (5.33)$$

The derivation can be found in Appendix B, p. 151. It follows mainly that of the recursive least square algorithm [129, 130]. The EKF recursions for a neural network are:

$$\mathbf{K}(n) = \lambda^{-1} \mathbf{P}(n-1) \mathbf{J}(n) \left[ \mathbf{I} + \lambda^{-1} \mathbf{J}^T(n) \mathbf{P}(n-1) \mathbf{J}(n) \right]^{-1} , \quad (5.34)$$

$$\hat{\mathbf{w}}(n) = \hat{\mathbf{w}}(n-1) + \mathbf{K}(n) (\mathbf{t}(n) - \mathbf{y}(\hat{\mathbf{w}}(n-1), \mathbf{x}(n))) , \quad (5.35)$$

$$\mathbf{P}(n) = \lambda^{-1} \mathbf{P}(n-1) - \lambda^{-1} \mathbf{K}(n) \mathbf{J}^T(n) \mathbf{P}(n-1) , \quad (5.36)$$

where  $n$  is the number of iterations,  $\mathbf{K}(n)$  is the so-called Kalman gain matrix,  $\mathbf{P}(n)$  is the state covariance matrix,  $\mathbf{J}(n)$  is the Jacobi matrix and  $\mathbf{I}$  is the identity matrix. In order to proceed in the minimisation we need to cycle through these recursively defined equations. The extended Kalman filter allows us to perform on-line learning, i.e. the weight vector can be updated after presentation of each example, which is superior to off-line learning, e.g. in the Gauss-Newton algorithm.

It is informative to compare the extended Kalman filter recursions with the update formulas in second order algorithms of the previous section. Let us rewrite the update of the weight vector in the EKF in equation (5.35) as:

$$\Delta \mathbf{w}(n) = \eta(n) \mathbf{K}(n) , \quad (5.37)$$

with  $\eta(n) = \mathbf{t}(n) - \mathbf{y}(\hat{\mathbf{w}}(n-1), \mathbf{x}(n))$ . For the Kalman gain matrix we are able to write:

$$\mathbf{K}(n) = \rho(n) \mathbf{P}(n-1) \mathbf{J}(n) , \quad (5.38)$$

where  $\rho(n) = \lambda^{-1} \left[ \mathbf{I} + \lambda^{-1} \mathbf{J}^T(n) \mathbf{P}(n-1) \mathbf{J}(n) \right]^{-1}$ . The inverse in the definition of  $\rho$  is just a factor for a network with only one output node. From these two equations the weight update in the EKF looks now as follows:

$$\Delta \mathbf{w}(n) = \gamma(n) \mathbf{P}(n-1) \mathbf{J}(n) , \quad (5.39)$$

with  $\gamma(n) = \eta(n) \rho(n)$ . If we compare this update with the Quasi-Newton update:

$$\Delta \mathbf{w} = \eta \left( \hat{\mathbf{H}}(\mathbf{w}) \right)^{-1} \mathbf{J}(\mathbf{w}) , \quad (5.40)$$

it is evident, that the EKF is essentially a Quasi-Newton optimisation algorithm, but now for *on-line* learning. The inverse of the matrix  $\mathbf{P}(n)$  in the EKF is defined as:

$$\mathbf{P}^{-1}(k) = \sum_{p=1}^k \mathbf{J}(p)\mathbf{J}^T(p) \lambda^{k-p} . \quad (5.41)$$

The state covariance matrix  $\mathbf{P}(n)$  is expressed as a weighted history of Gauss-Newton approximations to the inverse Hessian, see eq.(5.26). The diagonal elements determine the strength of the update step and can be viewed as approximations of the uncertainty in the current weight estimates.

In order to reduce the computational burden imposed by updating the parameter vector after presentation of each example and rather not only once at the end of one epoch, an *adaptive EKF* algorithm has been proposed [127]. The reduction is achieved by setting a threshold, defined as a fraction of the current root mean squared error, on the filter update step. Only when the output error exceeds this threshold, the update is performed. Through this adaptive filtering, the number of unproductive steps can be greatly reduced.

In fitting potential-energy surfaces some parts of the surface may be more important than others, e.g. the region where a bond breaks. We have changed the original recursions of the extended Kalman Filter slightly to allow for a separate weighting of each example. The corresponding equations can be found in Appendix B, p. 151.

Furthermore, if information about the forces of the moving nuclei is obtainable, it may be advantageous to take account of this information. Since the neural network output function is an analytical representation of the inputs, its derivatives with respect to the inputs can be computed, see Appendix C, p. 157. In order to minimise the output of the net along with its derivatives, we have imposed the following cost function:

$$\epsilon(n) = \sum_{j=1}^n \alpha_j \left[ \|\mathbf{e}(j)\|_2 + \sum_{i=1}^N \left\| \frac{\partial \mathbf{t}(j)}{\partial x_i(j)} - \frac{\partial \mathbf{y}(\mathbf{w}, \mathbf{x}(j))}{\partial x_i(j)} \right\|_2 \right] \lambda^{n-j} , \quad (5.42)$$

where  $n$  is the number of iterations,  $N$  is the number of input nodes,  $(\mathbf{x}(j), \mathbf{t}(j))$  is the sequence of input/output patterns,  $\alpha_j$  is a weighting parameter for each example,  $\mathbf{e}(j)$  is the sequence of errors and  $\mathbf{y}(\mathbf{w}, \mathbf{x}(j))$  is the nonlinear output function of the neural network. If we minimise this error function, we obtain the following EKF recursions:

$$\mathbf{K}(n) = \alpha(n) \lambda^{-1} \mathbf{P}(n-1) \mathbf{J}(n) \left[ \mathbf{I} + \lambda^{-1} \mathbf{J}^T(n) \mathbf{P}(n-1) \mathbf{J}(n) \right]^{-1} , \quad (5.43)$$

$$\begin{aligned} \hat{\mathbf{w}}(n) &= \hat{\mathbf{w}}(n-1) + \mathbf{K}(n) (\mathbf{t}(n) - \mathbf{y}(\hat{\mathbf{w}}(n-1), \mathbf{x}(n))) \\ &+ \sum_{i=1}^N \alpha(n) \mathbf{P}(n) \frac{\partial \mathbf{J}(n)}{\partial x_i} \left( \frac{\partial \mathbf{t}(j)}{\partial x_i} - \frac{\partial \mathbf{y}(\mathbf{w}, \mathbf{x}(j))}{\partial x_i} \right) , \end{aligned} \quad (5.44)$$

$$\mathbf{P}(n) = \lambda^{-1} \mathbf{P}(n-1) - \lambda^{-1} \mathbf{K}(n) \mathbf{J}^T(n) \mathbf{P}(n-1) . \quad (5.45)$$

For the complete derivation of these equations please refer to Appendix B, p. 151. The update of the weight vector in (5.44) is now influenced by the error of the output and its derivatives.

## 5.7 Generalisation

When training a neural network, one does not really want to reach full convergence to the minimum but is rather interested in obtaining a network with optimal *generalisation* performance. Generalisation means that the network should not only represent the fitted data set very well but is also able to predict new data points reliably.

During optimisation often *overfitting* occurs. This refers to the situation that while the network seems to get better and better and the error on the training data set decreases, the error on unseen examples increases as plotted in Fig. 5.8, i.e. globally the network is getting worse again.

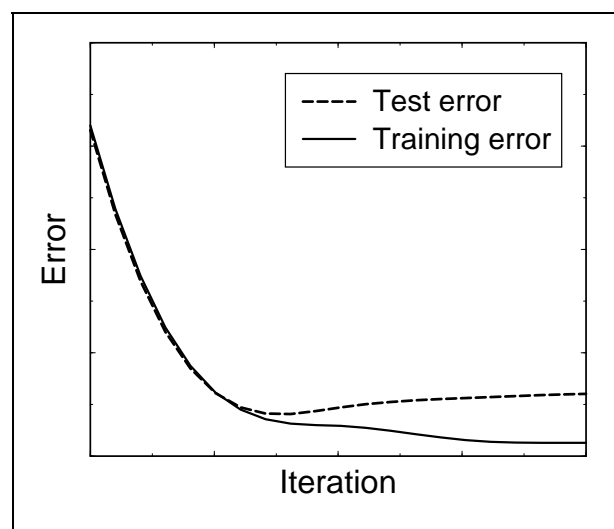


Figure 5.8: *Evolution of the error for a typical neural network. The error of the trained data set decreases continuously, whereas the error of unseen examples, the test error increases at some point. This behaviour is called overfitting.*

Typically the generalisation error is estimated by a *validation* or *test error*, i.e. the average error on a validation or test set, a fixed set of examples not present in the training set. In order to improve the generalisation of the network the most commonly used technique is *early stopping*:

- Split the training data into a training and a test set.
- Train only on the training set but monitor the error on the test set during optimisation
- The error on the training set will decrease, whereas the error on the test set will first decrease and then increase.
- Stop training as soon as the error on the validation set is higher than it was before. It is here that the network weights provide the best generalisation.

Early stopping is widely used because it is simple to understand and implement. Besides *early stopping* one should also try to keep the dimensionality of the network as low as possible in order to avoid extreme overfitting.

## 5.8 Regularisation

In order to achieve good generalisation with neural networks overfitting needs to be controlled. This can be achieved by early stopping, as previously discussed, or also by an application of weight penalty factors or regularisation parameters, i.e. a constraint is put on the parameters of the network. That is, instead of only minimising the usual root mean squared cost function, we augment it with a regularisation  $\mu R(\mathbf{w})$ :

$$E(\mathbf{w}) = \frac{1}{N} \sum_{n=1}^N [\mathbf{t}(\mathbf{x}(n)) - \mathbf{y}(\mathbf{w}, \mathbf{x}(n))]^2 + \mu R(\mathbf{w}) , \quad (5.46)$$

where  $R(\mathbf{w})$  is a function of the network parameters. Regularisation is usually performed by *weight decay* [52]. Since overfitting occurs when weights of the network become too large, one uses a penalty term in the cost function for large parameters:

$$E(\mathbf{w}) = \frac{1}{N} \sum_{n=1}^N [\mathbf{t}(\mathbf{x}(n)) - \mathbf{y}(\mathbf{w}, \mathbf{x}(n))]^2 + \mu \|\mathbf{w}\|^2 . \quad (5.47)$$

With weight decay the network itself removes non-useful connections during training by giving each connection a tendency to decay to zero, so that they disappear unless reinforced. The simplest way to apply weight decay is to use a scalar weight decay parameter  $\mu$ , but also more sophisticated approaches exist [131, 132]. The extended Kalman filter can be viewed as a general case of the weight decaying method, when initialised with weights close to zero [133]. Therefore, we need to apply weight decay only to optimisation schemes like backpropagation and conjugate gradients.

## 5.9 Normalising the data

Gradient-descent based minimisation algorithms like backpropagation can be very slow, particularly for a multi-layer neural network where the cost function is typically non-quadratic, non-convex, and high dimensional with many local minima and flat regions. [54]. It is therefore recommended to pre-process the input and output data.

### Normalising the inputs

Convergence is usually faster if the average of each input variable over the training set is close to zero. The reason is that a nonzero mean in the input variables creates a very large eigenvalue of the Hessian matrix. This means, the condition number will be large, i.e. the cost surface will be steep in some directions and shallow in others so that convergence will be very slow. Therefore it is recommended to preprocess the inputs by subtracting their means. This is also the reason why the *tanh*-function usually converges better than the sigmoidal function. The hyperbolic tangent is symmetric about the origin and is therefore more likely to produce outputs, which are also inputs to the next layer, that are on average close to zero.

Convergence is also faster if the inputs are scaled so that all have about the same covariance  $C_i$ :

$$C_i = \frac{1}{P} \sum_{p=1}^P (z_i^p)^2, \quad (5.48)$$

where  $P$  is the number of training examples,  $C_i$  is the covariance of the  $i^{\text{th}}$  input variable and  $z_i^p$  is the  $i^{\text{th}}$  component of the  $p^{\text{th}}$  training example. Scaling helps because it balances out the rate at which the weights learn. Inputs that have a large variation in spread along different directions of the input space will have a large condition number and slow learning.

### Normalising the outputs

If one uses a non-linear activation function in the output layer, then it is necessary to normalise the output variables in order to match the output range of the units. However, it is not recommended to set the maximum and minimum target values at the value of the activation functions asymptotes. One reason is that instabilities can result. The training process will try to drive the output values of the units as close as possible to the target values, which can only be achieved asymptotically. As a result, the weights are driven to larger and larger values, where the derivative of the activation function is close to zero. The weights may become stuck. A solution to this problem is to set the target values to be within the range of the activation functions, rather than at their asymptotic values. But care must be taken to insure that the node is not restricted to only the linear part of the activation function. Setting the target values to the point of the maximum second derivative on the transfer function is the best way to take advantage of the nonlinearity without saturating the activation function.

## 5.10 Weight initialisation

The starting values of the parameters can have a significant effect on the learning process. The weights in the neural network should be chosen randomly but in such a way that the transfer function is activated in its linear region. Taking the weights in this intermediate range has the advantage that learning can proceed, because the gradients are large enough, and the activation function of each node will be far away from the region of saturation. However, achieving this means to coordinate between the normalisation of the data, the choice of the transfer function and the choice of the weight initialisation. Having normalised the inputs and outputs as described in the previous section, the weights should be randomly drawn from a uniform distribution with mean zero and standard deviation, where  $m$  is the number of connections reaching into the specific node [54]:

$$\sigma_w = m^{-\frac{1}{2}}. \quad (5.49)$$

## 5.11 Ill-conditioning

The minimisation of the neural network cost-function is, more often than not, an ill-posed problem, i.e. it does not meet the following requirements:

- The network can learn the desired function, i.e. a solution exists.
- The solution is unique.
- The solution is stable under small variations in the training data set.

The first requirement can be achieved by using many hidden nodes, since any continuous function can be constructed with a single hidden layer with sigmoidal functions. Furthermore, we may be happy with any solution and ignore the second requirement and questions on uniqueness. However, a network that has learned the training data perfectly will be very sensitive to changes in the training data, the third requirement. A solution which changes significantly with slightly different training sets will have poor generalisation properties. This problem can be solved by applying early-stopping and regularisation techniques like weight-decay as discussed before.

However, an important problem for gradient-based learning methods is the particular form of the error function that represents the learning problem. It is well known that the derivatives of the error function are usually ill-conditioned. This ill-conditioning is reflected in error landscapes which contain many saddle points and flat areas where the Hessian matrix is bad conditioned. It appears that feed-forward learning tasks are generally characterised by having a singular or near-singular Hessian [134], which leads to inaccuracies due to the limited floating point accuracy of the digital computer. One reason is the ill character of the incorporated transfer function, the sigmoidal functions. They saturate at larger input values leading to nearly-zero derivatives and therefore singularity. Also, bad conditioning can be the result of uncentred data, which can be alleviated by normalising the inputs and outputs as already discussed.

However, there is another important reason for singularity stemming from the structure of the feed-forward network. When a multi-layer network has a small weight leaving from a hidden unit, the influence of the weights that feed into this hidden unit is significantly reduced. Due to the chain rule the derivative will be close to zero, leading to near-null rows in the Jacobi matrix and a near-singular Hessian. This kind of singularity is very common in neural network learning: The gradients in the lower-layer weights are influenced by the higher-layer weights.

One should keep in mind that ill-conditioning is a characteristic problem in feed-forward network learning.

## 5.12 Fortran program

In the scope of this work we have written a Fortran90 multi-layer feedforward neural network program. The results we will present throughout the following chapters have been obtained using this computer program.

The main features of the neural network program are:

- adjustable number of hidden layers,
- adjustable number of nodes in each layer,
- direct links between input and output nodes possible,
- random weight initialisation,
- flexible choice of activation functions for hidden and output layer,
- implemented activation functions:
  - linear functions,
  - sigmoid function,
  - hyperbolic tangent function,
  - sine functions,
  - Gaussian-like functions,
- random choice of training examples,
- optimisation algorithms:
  - adaptive backpropagation with momentum
  - conjugate gradients
  - adaptive global extended Kalman filter
- weighting of the input data set can be applied,
- “forces” can be included in the minimisation,
- weight decay,
- pre-conditioning of the data set.

The neural network program can be used in two different modes, the training and the production mode. In the first one the minimisation of the cost function is performed, whereas in the latter the already trained network can be used to produce new data, e.g. in molecular dynamics or for the plotting of iso-surfaces through the configuration space of the network function.





---

## Chapter 6

### *Ab initio* molecular dynamics

---

In *ab initio* on-the-fly molecular dynamics the forces acting on the molecule are determined by density functional theory. One approach is to first solve the Kohn-Sham equations self-consistently to obtain the electronic ground state and the forces of the nuclei as described in Chapter 3. In a second step these forces are used to integrate the equations of motion for the nuclei for the next time step (see, e.g., Bockstedte et al. [11]). Car and Parrinello followed a different approach to *ab initio* MD [135]. They treat the electronic and nuclear degrees of freedom together *at the same time* by simulating a fictitious classical dynamical system. Both approaches to *ab initio* MD simulations are accurate and do not require any fitting of the PES. However, since for each step the forces are determined by a new DFT calculation, they are very time-consuming. Their application is limited to problems where a small number of trajectories is sufficient to gain insight into a particular process, e.g. for an investigation of the adsorption and desorption mechanism of molecules on semiconductor surfaces [39].

However, for a theoretical description of sticking probabilities of molecules on metal surfaces we need to consider a good statistical average over the different initial configurations of the molecule approaching the surface [40]. This may require  $10^6 - 10^7$  evaluations of the potential and the forces. *Ab initio* on-the-fly molecular dynamics simulations are not applicable to problems with extensive statistics. Therefore Groß and Scheffler proposed a “divide and conquer” approach for *ab initio* molecular dynamics which is built on three independent steps [40]: first, one determines the *ab initio* PES by DFT. In a second step an interpolation between the actually calculated *ab initio* energies is performed. The last step consists of a molecular dynamics calculation on this continuous representation of the *ab initio* PES. Throughout the following we will assume that we have solved the problem of evaluating the potential energies, have furthermore developed a suitable model-form for the PES, and now wish to determine the final arrangement of the molecules given their initial configuration and momenta. This will be done by performing MD calculations on the continuous Neural Network potential-energy surface.

Since atoms and molecules are quantum particles, reaction probabilities should in principle be determined quantum mechanically, i.e. by solving the time-dependent or time-independent Schrödinger equation. However, these calculations are very time- and/or memory-consuming. It is often sufficient to perform classical molecular dynamics by solving Newton’s or Hamilton’s equations of motion.

## 6.1 Quantum dynamics

One common approach in quantum dynamics is to solve the time-independent Schrödinger equation:

$$[ H(\{\mathbf{R}\}) - E ] \Psi(\{\mathbf{R}\}) = 0. \quad (6.1)$$

Here  $H(\{\mathbf{R}\})$  is the Hamilton operator for the *nuclei* of the system, where the potential energy entering the Hamiltonian is the DFT energy, and  $\{\mathbf{R}\}$  are the ionic coordinates. The corresponding wave function  $\Psi$  needs to be expanded in a finite basis set. The problem is that even for the six-dimensional model of hydrogen dissociation on metal surfaces up to 21000 basis functions, so-called channels, need to be employed in the calculation. Therefore six-dimensions are currently the cutting-edge for theoretical quantum dynamics of reactions on surfaces [41, 40]. The movement of substrate atoms which is important for the description of the desorption of  $\text{H}_2/\text{Si}$  is so far not possible to be incorporated in quantum dynamical calculations.

The same holds for time-dependent or wave-packet dynamics where the time-dependent Schrödinger equation is solved:

$$i\hbar \frac{\partial}{\partial t} \Psi(\mathbf{R}, t) = H \Psi(\mathbf{R}, t). \quad (6.2)$$

Here the wave function and the potential needs to be described on a finite grid of points. The number of points becomes rather large with increasing dimension and therefore limits the application to problems with up to six degrees of freedom [63, 64].

Quantum calculations for hydrogen dissociation on metal surfaces have proven that indeed quantum effects play a role in this dissociation process. The initial sticking probability of  $\text{H}_2$  molecules impinging at the clean Pd(100) surface has been predicted to have an oscillatory structure, reflecting the quantum nature of the scattering process [41, 40, 136]. However, up to now these oscillations have not been detected in molecular beam experiments due to incoherence effects and a finite temperature and resolution of the experimental setup [137, 138]. In a study of the dissociative adsorption of hydrogen on a sulfur covered Pd(100) surface it has been found that quantum calculations tend to shift the sticking curve to lower results in comparison to classical calculations [51]. The reason is that the interaction of a  $\text{H}_2$  molecule with the substrate gives rise to zero-point energies and this energy needs to be taken from the incident beam and therefore the effective kinetic energy of the impinging molecule is reduced. Consequently the fraction of the molecules which are able to overcome a certain barrier towards dissociation is reduced and the sticking probability is lower. However, the overall trend of the sticking curve with increasing kinetic energy is well reproduced by classical dynamics. Furthermore, in the case of heavier molecules like CO the quantum effects should not play a significant role due to the larger mass and the smaller de Broglie wavelength of the CO molecule. For these reasons classical molecular dynamics calculations give results that are qualitatively and even semi-quantitatively in agreement with molecular beam experiments for the adsorption of molecules on surfaces.

## 6.2 Classical dynamics

The classical-dynamics calculations are performed by solving the Hamilton equations of motion:

$$\dot{q}_i = \frac{\partial H}{\partial p_i}, \quad (6.3)$$

$$\dot{p}_i = -\frac{\partial H}{\partial q_i}, \quad (6.4)$$

where  $H$  is the Hamiltonian for the nuclei of the system and the  $q_i$  and  $p_i$  are the generalised coordinates and momenta, respectively. The equations of motion can be numerically integrated with standard integration schemes, like Runge-Kutta or the Burlish-Stoer method [124]. There is no problem of incorporating higher dimensions into the dynamics since no expansion of a wave function into basis functions or representation on a grid is requested.

The Hamiltonian of a hydrogen molecule interacting with a rigid surface, i.e. moving on a PES depending on the six degrees of freedom of the hydrogen molecule  $V(\mathbf{R}, \mathbf{r})$ , can be written as:

$$H = \frac{1}{2M} \nabla_{\mathbf{R}}^2 + \frac{1}{2\mu} \nabla_{\mathbf{r}}^2 + V(\mathbf{R}, \mathbf{r}), \quad (6.5)$$

where  $\mathbf{R} = (\mathbf{r}_1 + \mathbf{r}_2)/2 = (X_c, Y_c, Z_c)$  and  $\mathbf{r} = (\mathbf{r}_2 - \mathbf{r}_1)$  are the centre-of-mass and relative coordinates of the hydrogen molecule,  $M = 2m$  and  $\mu = m/2$  are the total and reduced mass of the hydrogen molecule with  $m$  the mass of one hydrogen atom. Writing the relative part of the Hamiltonian in spherical coordinates gives:

$$H = \frac{1}{2M} (p_{X_c}^2 + p_{Y_c}^2 + p_{Z_c}^2) + \frac{1}{2\mu} \left( p_r^2 + \frac{p_\theta^2}{r^2} + \frac{p_\phi^2}{r^2 \sin^2 \theta} \right) + V(\mathbf{R}, \mathbf{r}). \quad (6.6)$$

Inserting this Hamiltonian into Hamilton's equations of motion, see equations (6.3) and (6.4), we get the following equations:

$$\begin{aligned} \dot{\phi} &= \frac{1}{\mu} \frac{p_\phi}{r^2 \sin^2 \theta}, & \dot{p}_\phi &= -\frac{\partial V}{\partial \phi}, \\ \dot{\theta} &= \frac{1}{\mu} \frac{p_\theta}{r^2}, & \dot{p}_\theta &= \frac{1}{\mu} \frac{p_\phi^2 \cos \theta}{r^2 \sin^2 \theta} - \frac{\partial V}{\partial \theta}, \\ \dot{r} &= \frac{1}{\mu} p_r, & \dot{p}_r &= \frac{1}{\mu} \frac{p_\theta^2}{r^3} + \frac{1}{\mu} \frac{p_\phi^2}{r^3 \sin^2 \theta} - \frac{\partial V}{\partial r}, \\ \dot{X}_c &= \frac{1}{M} p_{X_c}, & \dot{p}_{X_c} &= -\frac{\partial V}{\partial X_c}, \end{aligned}$$

$$\begin{aligned}\dot{Y}_c &= \frac{1}{M} p_{Y_c}, & \dot{p}_{Y_c} &= -\frac{\partial V}{\partial Y_c}, \\ \dot{Z}_c &= \frac{1}{M} p_{Z_c}, & \dot{p}_{Z_c} &= -\frac{\partial V}{\partial Z_c},\end{aligned}$$

In this work the equations of motion are numerically integrated with the Burlish-Stoer method with a variable time step [124].

We point out that there is no unambiguous definition of the sticking probability in the calculation. Because we do not account for energy dissipation to the substrate, after the bond-breaking the two hydrogen atoms will continue to travel on the surface. In reality, the atomic fragments will transfer their kinetic energy to substrate excitations like electron-hole pairs or phonons and come to rest on the surface. However, especially in the case of light molecules such as hydrogen dissociating on metal surfaces the energy transfer to the substrate is very small due to the large mass mismatch. Whether a molecule sticks on the surface or not is almost entirely determined by the bond-breaking process. In our calculations we will therefore count the molecule as dissociated on the surface if the bond length between the two atoms is significantly stretched ( $\geq 300\%$ ) and the two fragments remain close to the surface.

Another important point is the sampling of the initial configurations. Sticking corresponds to a process in which statistically distributed particles hit the surface. Therefore the determination of classical sticking probabilities requires an average over typically thousands of trajectories. In this work we chose the initial conditions by some Monte-Carlo sampling and stop the calculation when convergence of the sticking probability as a function of the number of initial configurations is achieved.

Furthermore we will restrict ourself to calculations of the sticking probabilities for normal incidence of the molecule, i.e. with a momentum perpendicular to the substrate.

**Part II.**  
**Tests: Neural Network fits to  
analytical functions**



---

## Chapter 7

### Neural Network tests: Simple analytical functions

---

As a first test of the generalisation and interpolation ability of neural networks (NN) we studied simple low dimensional analytical functions. We will present results for the approximation of trigonometric functions in one and two dimensions and the one dimensional harmonic oscillator based on a discrete set of points. These functions are often employed in analytical fits to potential-energy surfaces, e.g. for the dissociation process of hydrogen on Pd(100) [41].

Before doing interpolations with neural networks it is necessary to choose the structure of the network and the optimisation algorithm. For the basic test problems presented in this chapter any minimisation routine, like backpropagation, conjugate gradients or the adaptive extended Kalman filter can be used with equal success. Their performance and the application of different activation functions in the neural network context will be discussed in detail for a more difficult and realistic six-dimensional problem in the next chapter. Throughout the following we will always use the adaptive extended Kalman filter as the optimisation method, hyperbolic tangents as activation functions in hidden nodes and a linear function in the output layer.

If information about the value of the desired function and its derivative at a discrete number of points is available simultaneously, it might be advantageous to employ this knowledge in the interpolation. We have rewritten the extended Kalman filter equations to include the derivatives along with the value of the function in the error function as explained in section 5.6.5, p. 51. Since the derivatives of analytical functions can be calculated in a straightforward manner, we will test also their influence on the neural network performance.

#### 7.1 Trigonometric function (1D)

A basic one dimensional function, which is often used in analytical fits of potential-energy surfaces, is the trigonometric sine function:

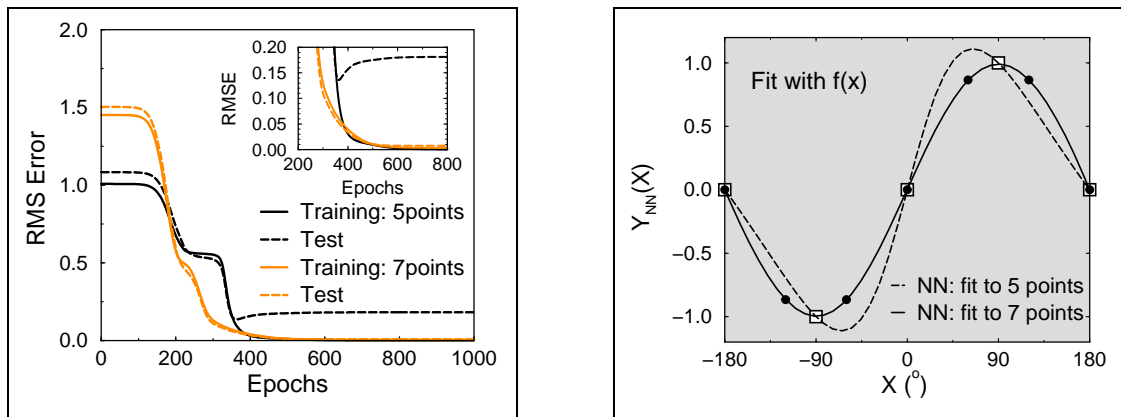
$$f(x) = \sin(x) . \tag{7.1}$$

The interval  $[-180^\circ, 180^\circ]$  has been divided into a set of three, five and seven points.

### Neural network fit with values only

We chose a  $1-2-1$   $tl$  neural network for the interpolation, i.e. with one input node representing the values of  $x$ , one hidden layer with two hyperbolic tangent nodes  $t$  and one linear output node  $l$  with the activation function  $f(x) = x$ . Accordingly, the network consists of seven parameters, four connecting the input plus the bias node with the hidden nodes and three from the hidden nodes as well as the bias to the single output node (for an illustration see Fig. 5.2, p. 44).

We have plotted the development of the root mean squared error (RMSE) as a function of the number of presentations of the whole data set, called an *epoch*, in Fig. 7.1(a). The neural network interpolation is based on five or seven points obtained from the analytical function. The test set consists of 100 points in the interval  $[-180^\circ, 180^\circ]$ . The training error drops fast and levels off around 600 epochs for both fits. However, the error on the test set for the fit with five points starts to grow again around 400 epochs, the network is not able to generalise well. This can be seen by looking at the neural network output in Fig. 7.1(b). The network fit to five points of the sine function approximates the given values of the function very well, however the maxima and minima have a higher value and their location is shifted. With two more points included in the fit the sine function is well represented. Test and training error are below 1%. The runtime of the NN-program on a IBM RISC 6000 machine is a few seconds.



(a) Training and test set error

(b) Neural Network output function

Figure 7.1:  $1-2-1$   $tl$  Neural Network fit of  $f(x) = \sin(x)$ . The training set consists of either 5 or 7 points, the test set corresponds to 100 points in the interval  $[-180^\circ, 180^\circ]$ . One epoch is the presentation of all training points. The Kalman filter parameters are  $\lambda(0) = 0.98$ ,  $\lambda_0 = 0.99926$  and the adaptive threshold  $\nu = 0.3$  RMSE.

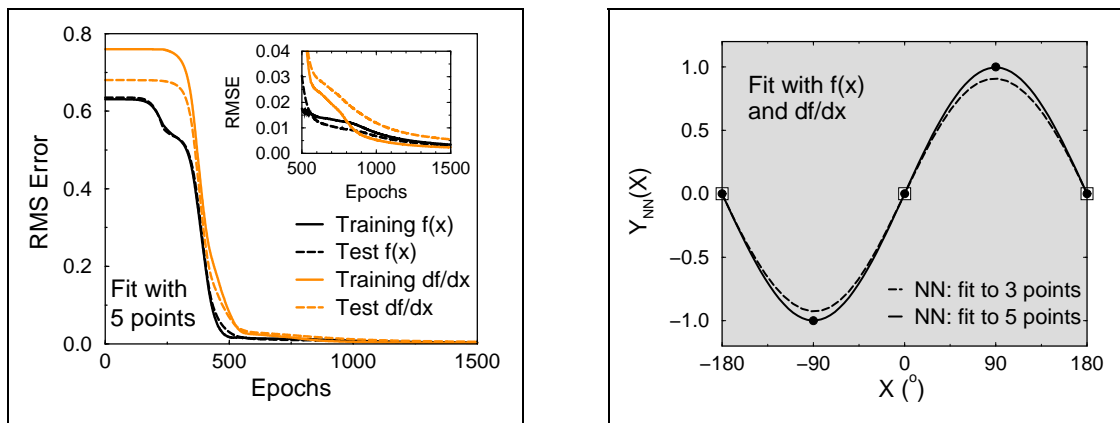
### Neural Network fit with values and derivatives

The derivatives of the desired function - if available - can be approximated along with the value of the function by minimising the following error function:

$$\epsilon = \epsilon(f) + w_{f'} \epsilon(f'), \quad (7.2)$$



where  $\epsilon(f)$  is the error corresponding to the value of the function and  $\epsilon(f')$  is the error connected to the derivatives. The factor  $w_{f'}$  can be used to switch on or off the information about the derivatives. We allowed the neural network to have greater flexibility in correspondence to the higher information available through the derivatives and used a  $1-3-1$  *tl* network with 10 parameters. The training and test set error in Fig. 7.2(a) for a neural network fit of the sine function with 5 function values and 5 derivatives continues to go down even at 1500 epochs without showing any signs of overfitting. Hence, the use of the derivatives in the extended Kalman filter update equations allows to produce a perfect fit with only five points, whereas without the derivative seven points were required to achieve equal accuracy. Even with three points, see Fig. 7.2(b), a qualitatively good approximation can be obtained.

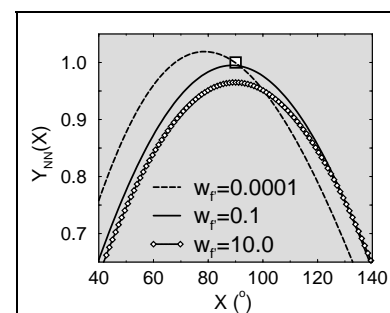


(a) NN errors for a fit to 5 points

(b) NN output for fits to 3 and 5 points

Figure 7.2: Neural network fit of  $f(x) = \sin(x)$  with information on the value and the derivative. The training set consists of three and five points, the test set corresponds to 100 points in the interval  $[-180^\circ, 180^\circ]$ . The Kalman filter parameters are  $\lambda(0) = 0.98$ ,  $\lambda_0 = 0.99926$  and the adaptive threshold  $\nu = 0.3$  RMSE.

The factor  $w_{f'}$  in equation (7.2) can be used to weight the influence of the derivative information in the cost function. With a higher factor than “1” the locations of the maxima are perfectly represented. A factor between  $0.1-1$  produces very well results on both the potential and the derivatives, whereas any lower value over-favours the value of the function. This can be seen from Fig. 7.3 where a zoom into the maximum of the sine function at  $x = 90^\circ$  has been plotted. It is sometimes necessary to set the weighting factor to a value lower than “1”, because the differences between the estimated and the desired derivatives can be significantly larger than the difference in the potential. This can lead to an exponential growth of the elements in the Kalman state covariance matrix and hence to numerical instabilities, see section 5.6.5, p. 51.



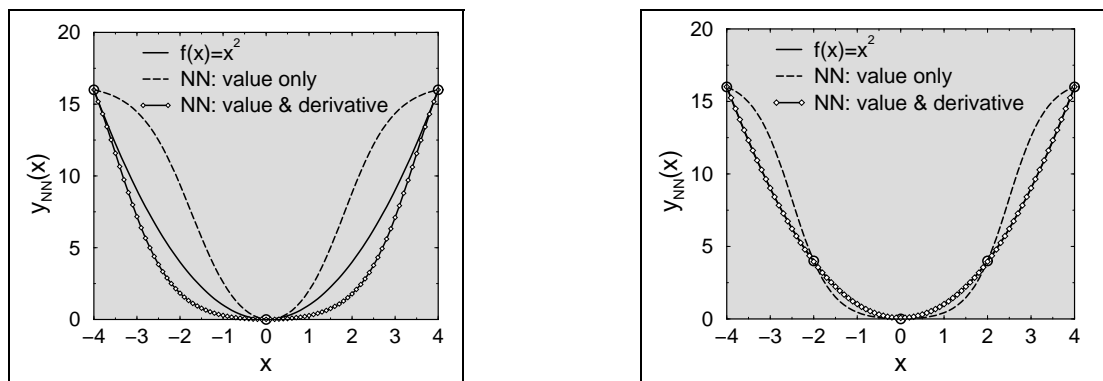
**Figure 7.3:** NN-output for different weighting factors  $w_{f'}$ . The white square marks the training point.

## 7.2 Harmonic Oscillator (1D)

A simple physical meaningful test function is the harmonic oscillator:

$$f(x) = x^2. \quad (7.3)$$

For the approximation we have divided the interval  $[-4, 4]$  into a set of three, five and seven points and performed a neural network fit with a  $1-2-1$   $tl$  net with seven parameters on the value of the harmonic oscillator only and on the value and the derivative  $\frac{df}{dx} = 2x$ . If we employ information on the value of the function only in the



(a) NN-fit with 3 points

(b) NN-fit with 5 points

Figure 7.4: Neural network fit to  $f(x) = x^2$  with 3 and 5 points in the interval  $[-4, 4]$ . The approximations have been performed with a  $1-2-1$   $tl$  net using either the value only or the value & the derivative. Parameter:  $\lambda(0) = 0.98$ ,  $\lambda_0 = 0.99946$  and  $w_f = 0.1$ .

optimisation procedure we need - as in the case of the one dimensional sine function - seven points, whereas with knowledge about the derivative five points are enough to produce a perfect fit. The neural network output function for the fits to three and five points can be seen in Fig. 7.4. With derivative information already three points give a qualitatively good approximation. The errors for the different interpolations have been listed in table 7.1. The best fits with minimum complexity have been marked bold.

Data #	Fit with value only				Fit with value and derivative			
	Training set		Test set		Training set		Test set	
	RMSE $f(x)$	RMSE $df/dx$	RMSE $f(x)$	RMSE $df/dx$	RMSE $f(x)$	RMSE $df/dx$	RMSE $f(x)$	RMSE $df/dx$
3	$5 \cdot 10^{-4}$	5.99	3.77	3.65	$1 \cdot 10^{-3}$	$8 \cdot 10^{-3}$	1.47	1.39
5	$2 \cdot 10^{-5}$	4.72	1.75	2.93	<b><math>5 \cdot 10^{-2}</math></b>	<b><math>2 \cdot 10^{-1}</math></b>	<b><math>4 \cdot 10^{-2}</math></b>	<b><math>9 \cdot 10^{-2}</math></b>
7	<b><math>6 \cdot 10^{-2}</math></b>	<b><math>6 \cdot 10^{-1}</math></b>	<b><math>1 \cdot 10^{-1}</math></b>	<b><math>3 \cdot 10^{-1}</math></b>	$3 \cdot 10^{-2}$	$5 \cdot 10^{-2}$	$3 \cdot 10^{-2}$	$5 \cdot 10^{-2}$

Table 7.1: RMS errors for the approximation of the harmonic oscillator with a  $1-2-1$   $tl$  network. Fits with information on the value only and with value and derivative data.

## 7.3 Trigonometric function (2D)

As a two dimensional test problem we took the function:

$$f(x) = \sin^2(x) + \cos(2y) . \quad (7.4)$$

We have chosen the fitting interval to be  $x \in [0^\circ, 360^\circ]$  and  $y \in [0^\circ, 180^\circ]$  in order to see if symmetry can be reproduced by the network. The test function is plotted in Fig. 7.5.

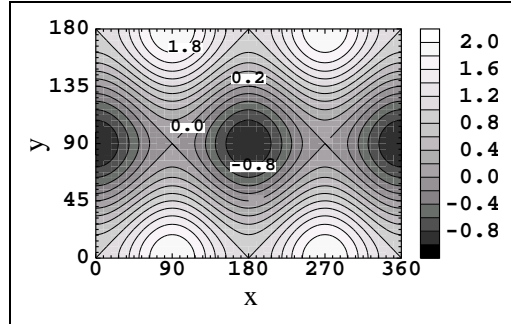
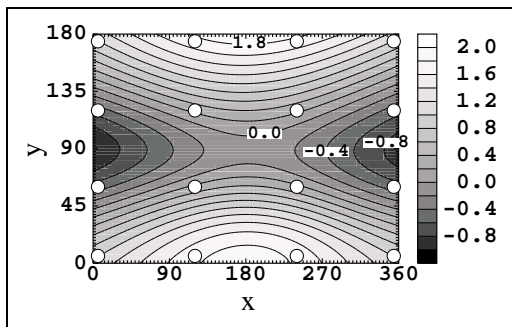


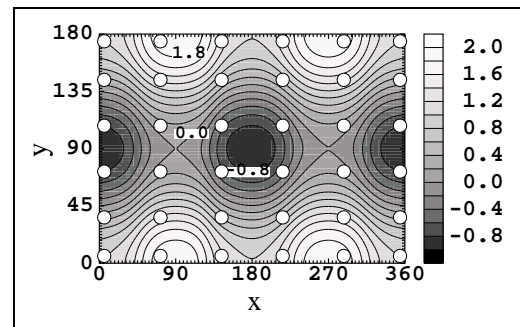
Figure 7.5: Neural Network test function :  $f(x) = \sin^2(x) + \cos(2y)$ .

### Neural network fit with values only

As different network architectures have been used, we will always present the best result for each training set. The neural network training set consist of 16, 25 and 36 points corresponding to a  $4 \times 4$ ,  $5 \times 5$  and  $6 \times 6$  grid in the two dimensions. The test set is created on a  $41 \times 41$  grid. Fitting the value of the function only requires to use a  $6 \times 6$  grid as plotted in Fig. 7.6(b), a lower number of points does not generate a satisfactory result, see Fig. 7.6(a). The given values of the functions in Fig. 7.6(a) are approximated well, but the symmetry is not reproduced.



(a) Neural Network fit with 16 points and a  $2-15-15-1$  tl network



(b) Neural Network fit with 36 points and a  $2-20-1$  tl network

Figure 7.6: Neural Network fit of the function  $f(x) = \sin^2(x) + \cos(2y)$  with information about the value of the function only. The different grids of data points are indicated by white circles in (a) and (b). Parameter:  $\lambda(0) = 0.98$ ,  $\lambda_0 = 0.9989$ .

### Neural network fit with values and derivatives

If we include again the information about the two derivatives  $\frac{\partial f}{\partial x} = 2 \sin(x) \cos(x)$  and  $\frac{\partial f}{\partial y} = -2 \sin(2y)$  in the fit procedure, already a  $4 \times 4$  grid gives good results as plotted in Fig. 7.7. From the previous examples we can conclude that the necessary number of points for a satisfactory approximation with a neural network can be reduced by at least two points per dimension if the fitted data set consists of values and derivatives of the desired function.

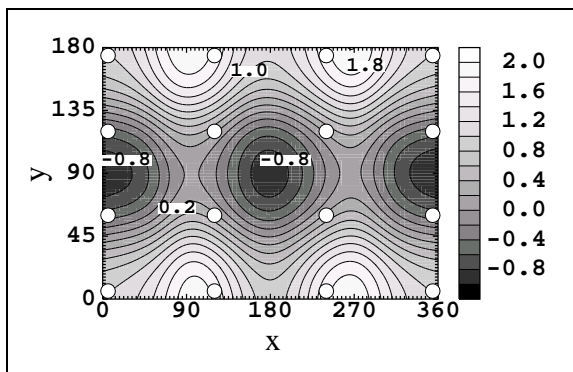


Figure 7.7: Neural Network fit with value and derivative information on a  $4 \times 4$  grid. The net is a 2–15–15–1 *tl* net. Training time is about 2 minutes on an IBM RISC 6000 machine.

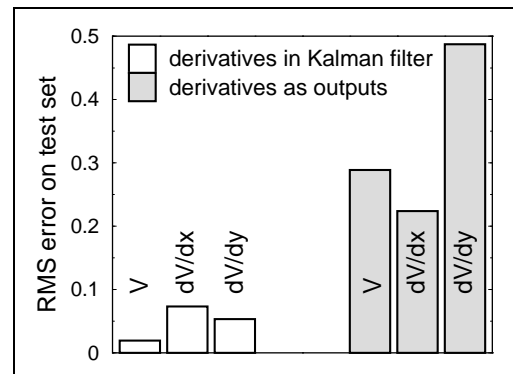


Figure 7.8: Test error for two different Neural Networks. The net with derivatives of the output node incorporated explicitly in the cost function has clearly smaller test errors.

Another approach of including the derivatives in the approximation is to use them explicitly as additional outputs of the neural network rather than incorporating them in the cost function. In our example this translates to a network with three output nodes, one for the value of the function  $f(x)$ , one for the derivative  $\frac{\partial f}{\partial x}$  and one for the other derivative  $\frac{\partial f}{\partial y}$ . We have tested this approach with an interpolation based on a grid of  $5 \times 6$  points. The *training errors* on the three different quantities for (a) a network with the derivatives as output nodes and (b) for a network with the derivative information incorporated in the cost function can be minimised in both cases to values below 1%. However, there is a significant difference between these two approaches which can be seen by looking at the error on the *test set* of points in Fig. 7.8. The generalisation ability of the network with the derivatives as additional outputs (grey bars) is clearly worse in comparison to the network with the derivative included in the Kalman filter update equations (white bars). This can be understood as follows. In the former case the network does not know that the partial derivatives of the first output, the value of the function, are equal to the values of the other outputs. At the beginning of the optimisation they are independent, their dependency needs to be learned. With our approach of using the partial derivatives of the output node representing the value of the desired function *explicitly* in the error function, we guarantee that the full knowledge is available from the start of the iterations which leads to clearly improved results.

---

## Chapter 8

### Neural Network test: 6-D analytical PES for H<sub>2</sub>/Pd(100)

---

In principle, neural networks can fit any continuous function with arbitrary accuracy. However, it is *a priori* unknown how many parameters and data points are necessary to obtain a good solution. We are interested in the adsorption probability of H<sub>2</sub> molecules on a potassium covered Pd(100) surface. Up to now, neural networks have not been applied in dynamical calculations of such reactions. In order to learn more about their practical approximation ability it is essential to find realistic test problems.

Analytical potential-energy surfaces for the sticking of H<sub>2</sub> on metal surfaces provide ideal test cases for the neural network approach for different reasons. First of all, the energy of an analytical PES is fast to evaluate. This allows us to study the influence of the sampling of the data points on the quality of the NN-approximation as measured by the root mean squared error (RMSE) in great detail. Secondly, analytical PESs have proven to describe such adsorption events realistically [26, 41, 51]. Furthermore, as an additional check of the accuracy of the obtained NN-model besides the RMS-error, we are able to compare the results of classical molecular dynamics (MD) calculations using the neural network representation to MD-calculations performed on the analytical PES. Namely, we can use the sticking probability - calculated with the neural network and the analytical PES - as a further test of the accuracy of the approximation.

The symmetry of the problem is not represented automatically by neural networks. In the mapping of PESs by DFT-calculations one takes excessive advantage of this symmetry by studying unit-cells and focusing on high-symmetric sites. We will discuss how the symmetry can be incorporated *explicitly* prior to the interpolation. In the last chapter we have seen that symmetry can also be *learned* by an *implicit* use of the forces during the optimisation. The application of the derivatives in the fitting of the simple test functions resulted only in a small reduction of training points per dimension. This effect will be even smaller with an *explicit* incorporation of the symmetry, because much of the information provided by the forces will be redundant due to symmetry constraints. Therefore and since the convergence of the forces in *ab initio* calculations takes a factor of 2–5 times longer than the convergence of the energy, we will not employ the forces in the approximation throughout the following. However, if they are available the developed neural network program is ready to use them.

## 8.1 *Ab initio* and analytical PES

The potential-energy surface of hydrogen dissociation on the clean palladium surface, H<sub>2</sub>/Pd(100), has been calculated by Wilke and Scheffler within density functional theory [29,22]. We discussed this PES already in the introduction for hydrogen dissociation on metal surfaces in Sec. 2.3, p. 17. The dissociation is non-activated, pathways to dissociation exist with no energy barrier, the molecule can freely dissociate above certain sites. The PES has been mapped out following the usual approach of calculating 2D cuts through the configuration space above high-symmetric geometries as presented in Sec. 3.3, p. 33. The equilibrium position of a hydrogen atom is the surface hollow site with a small adsorption height of 0.1 Å above the topmost palladium layer. The reaction is exothermic, hydrogen gains energy on adsorption. The minimum pathway for the dissociation of H<sub>2</sub> molecules is above the bridge site with the H-atoms oriented towards the hollow site.

The *ab initio* PES has been fitted with analytical functions by Groß et al. [41], which we described in Sec. 4.1, p. 36. The potential-energy surface is expressed as a function of the six degrees of freedom of the hydrogen molecule, the surface is kept rigid:  $V(X_c, Y_c, Z_c, d, \theta, \phi)$ , where  $X_c, Y_c$  and  $Z_c$  are the centre of mass coordinates of the hydrogen molecule,  $d$  is the distance between the two hydrogen atoms,  $\theta$  and  $\phi$  are the polar and azimuthal angles of the molecule. The potential in the  $Zd$  plane is described in reaction path coordinates  $s$  along the reaction path and  $r$  perpendicular to it [110, 41]. The fit has been performed by a least square method such that the difference between the analytical potential  $V(X_c, Y_c, s, r, \theta, \phi)$  and the *ab initio* total energies, which have been calculated for more than 250 configurations, on the average is smaller than 25 meV. In Fig. 8.1 two cuts through the six-dimensional configuration space of the analytical interpolation have been plotted. Fig. 8.1(a) shows the analytic

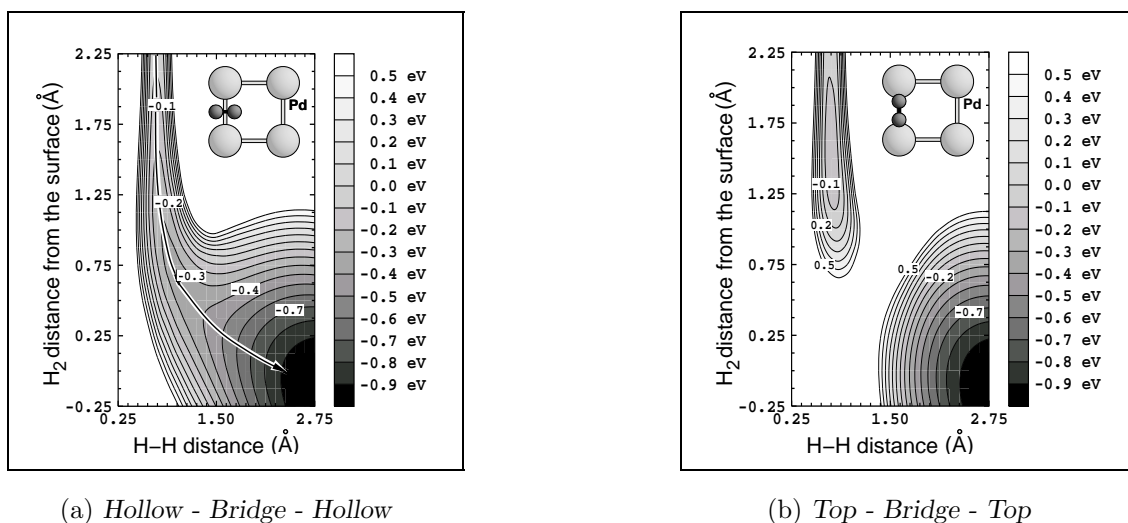


Figure 8.1: Contour plots through the 6-D analytical PES of H<sub>2</sub>/Pd(100) from [41]. Insets: Configuration of the dissociation pathways. The solid line in (a) marks the minimum path towards dissociation. The energy spacing of the contour lines is 0.1 eV.

interpolation of the minimum path which should be compared to the *ab initio* cut in Fig 2.1, p. 14. The solid line marks the dissociation pathway, it exhibits no barrier towards dissociation. However, if we turn the molecule around by  $90^\circ$ , keeping the molecular axis parallel to the surface, a distinct energy barrier of  $E_{barr} > 0.5$  eV exists (Fig 8.1(b)). Only one of the six coordinates has been changed and a qualitatively different dissociation behaviour of the molecule has been obtained. Both elbow plots differ only in the small region of the PES where the bond of the hydrogen molecule breaks. The entrance channels with the centre of mass of the molecule more than  $z = 1$  Å above the surface are very similar, as well as the exit channels with a molecular bond length  $r > 1.50$  Å. The crucial bond-breaking process of the molecule takes place in a relatively small region of the potential-energy surface. Throughout the following, we will often refer to these two-dimensional cuts through the configuration space, but one should always keep in mind that the complete PES is six-dimensional. In summary, the PES for the dissociation of hydrogen on the clean Pd(100) surface exhibits the following features:

- Minimum dissociation pathway:
  - Molecule dissociates with its molecular axis parallel to the surface.
  - Configuration: Centre of mass of the molecule over the bridge site with the atoms dissociating to the hollow site.
  - No energy barrier for dissociation.
- PES is strongly corrugated and anisotropic:
  - High energy barriers on-top sites.
  - No barrier for Pd bridge site, with the molecules oriented to the hollow site.
- Co-existence of activated and non-activated pathways, with the majority of paths being activated.

## 8.2 Tests of the Neural Network structure

In the following paragraphs we will check the influence of the choice of optimisation algorithms on the approximation ability and the training time of neural networks. We will then discuss the use of various activation functions and their performance for fitting potential-energy surfaces. We will further address the issue of ill-conditioning and the incorporation of symmetry constraints into neural networks before we summarise these results.

### 8.2.1 Optimisation algorithms

We have tested the performance of the backpropagation algorithm (BP), conjugate gradients (CG) and the extended Kalman filter (EKF) on a training set of 1560 data

points and 7200 test points obtained from the six-dimensional analytical PES. We choose a 6–16–16–1  $tl$  network with 401 parameters.

The backpropagation algorithm and the conjugate gradient method are designed for offline optimisation, i.e. the network weights are updated after a presentation of the *whole data set* to the neural network. However, backpropagation can also be used in online mode, i.e. when the parameters are being changed after the presentation of *each example* of the training set. The extended Kalman filter algorithm is especially designed for online minimisation. We have discussed these issues in Sec. 5.6, p. 48.

The evolutions of the test set errors with the number of epochs for the different optimisation schemes have been plotted in Fig. 8.2. The EKF algorithm is clearly superior to the other methods. It reaches a better minimum and shows a steep drop of the error function. One reason for its performance is the use of an approximation of the Hessian matrix, the matrix of second derivatives, for the weight update.

Furthermore, the improved results are due to the choice of *online* optimisation. Changing the weights after each example leads to a quick decrease of the error and increases the chances of not getting trapped in a local minimum at early stages of the optimisation. This is also supported by the fact that the BP minimisation exhibits an earlier drop of the error in online than in offline mode. It also reaches a slightly better minimum. The noise in the BP *online* update reflects that the algorithm it designed for *offline* mode. The CG method cannot compete with the EKF algorithm. It is an offline scheme using only the Jacobian matrix, the matrix of first order derivatives, for the parameter update. However, it leads to improved results in comparison to backpropagation in offline mode due to a more sophisticated weight update.

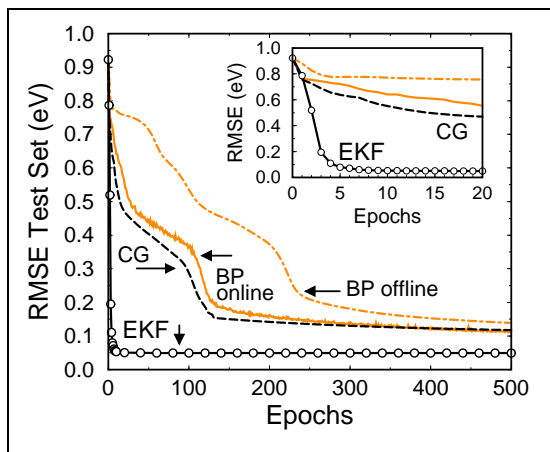


Figure 8.2: Root mean squared error on the test set for the fit to the 6D analytical PES of  $H_2/Pd(100)$  versus the number of epochs. The training set consists of 1560 examples. Inset: first 20 epochs. Network: 6–16–16–1  $tl$  net.

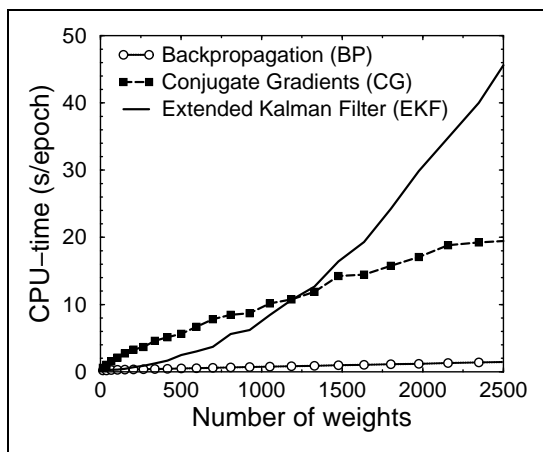


Figure 8.3: CPU-time versus number of weights of a neural network fit. One epoch is here the presentation of 500 examples to the neural network. The results have been obtained on one node of an IBM SP2 machine.

The superior performance of the Kalman filter algorithm has to be paid with longer training times of the neural network as plotted in Fig. 8.3. Since the EKF takes ad-



vantage of second order derivatives the training time as a function of the parameters, the weights, increases quadratically. The BP and the CG use only information on first derivatives and show therefore a linear behaviour. However, the calculation of the *ab initio* energies for a potential-energy surface takes several CPU years, whereas the fitting of these data using the EKF algorithm takes between minutes and hours, depending on the number of parameters and the size of the training set. The training time is thus just a fraction of the costs of the underlying DFT calculations and negligible.

### Adaptive Kalman filtering

The training time with the extended Kalman filter (EKF) can be reduced by applying an adaptive scheme: The weight update is only performed for those examples which exceed a certain threshold during each epoch [127]. The threshold can be defined as a fraction of the current root mean squared error. This algorithm is called the adaptive global extended Kalman filter (AGEKF). The percentage of the minimisation costs of the AGEKF compared to the EKF in Fig. 8.4 reveal that an adaptive threshold of  $0.3 \times RMSE$  reduces the training time to 80% and a value of 0.9 to 50% of the EKF's costs. The development of the test error with the number of epochs in Fig 8.5 drops less quickly for the different adaptive filter parameters than the error for the EKF. However, with factors up to a value of 0.9 one gets similar or even better results as compared to the EKF.

The adaptive filter helps to concentrate on the most valuable examples during each epoch and decreases the chance of getting stuck in a local minimum of the error surface too early during the minimisation. At the same time the AGEKF reduces the minimisation costs. We will therefore always apply the adaptive global extended Kalman filter for the optimisation of the network weights with adaptive parameters between 0.3 and 0.9.

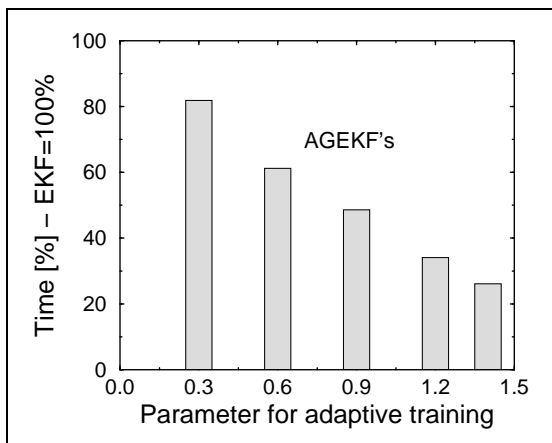


Figure 8.4: Time reduction with adaptive filtering compared to the EKF as a function of the adaptive parameter. A parameter of 0.3 implies an adaptive update threshold of  $0.3 \times RMSE$ .

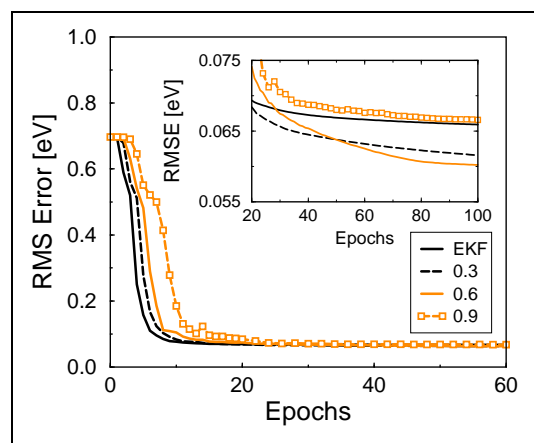


Figure 8.5: Adaptive Kalman filtering: Root mean squared error against the number of epochs for different adaptive parameters. Inset: zoom into the evolution of the errors.

### Kalman filter forgetting schedule

The forgetting schedule improves the approximation ability of the extended Kalman filter algorithm. It has been described in Sec. 5.6.5, p. 51. The neural network cost function is determined by summing the error over all previous iterations multiplied by a weighting factor  $\lambda(k)$  [127]. The time varying forgetting schedule changes the value of  $\lambda$  before the presentation of each training sample according to  $\lambda(k) = \lambda_0 \lambda(k-1) + 1 - \lambda_0$ , where  $\lambda_0$  is typically a constant between 0.99 and 0.9995 and the initial value of  $\lambda$ ,  $\lambda(0)$ , is chosen between 0.95 and 0.99. At the beginning of the minimisation the forgetting schedule is designed to take only the most recent examples into account for the weight update. This avoids too early trapping into local minima. When the process continues, all information available is used to improve the result.

The evolution of the training error with the number of epochs for a network with two different forgetting schedules is displayed in Fig. 8.6. The parameter set with the values  $\lambda(0) = 1.0$  and  $\lambda_0 = 1.0$  implies that all presented examples contribute to the cost function from the beginning of the iteration, i.e. the network is trained with full memory and without forgetting. It is clearly visible from Fig. 8.6 that this approach cannot avoid trapping into local minima of the error surface. In comparison, a forgetting schedule with parameters  $\lambda(0) = 0.98$  and  $\lambda_0 = 0.9994$  improves the results by orders of magnitude within 30 epochs. Smaller values of  $\lambda_0$  for the same  $\lambda(0)$  produce a less pronounced drop of the error, higher values can lead to numerical instabilities. The weight changes become very large, some activation functions get saturated and the Jacobi and Hessian matrix become singular. The change in the forgetting factor  $\lambda_0$  should be set so that controlled growth and collapse of the covariance matrix, the approximation of the Hessian matrix, occurs within 30 epochs of the training set. For a more detailed discussion please refer to Ref. [57].

Due to its strong influence on the obtained results it is recommended to perform a short number of iterations for different forgetting schedule parameters for each new problem. For potential-energy surfaces we obtained good approximations with an initialisation value of  $\lambda(0) = 0.98$  together with  $\lambda_0$  between 0.9982 and 0.9994.

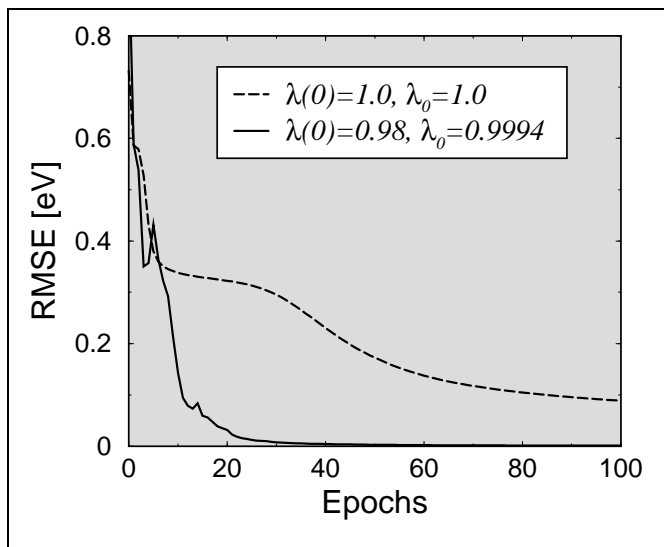


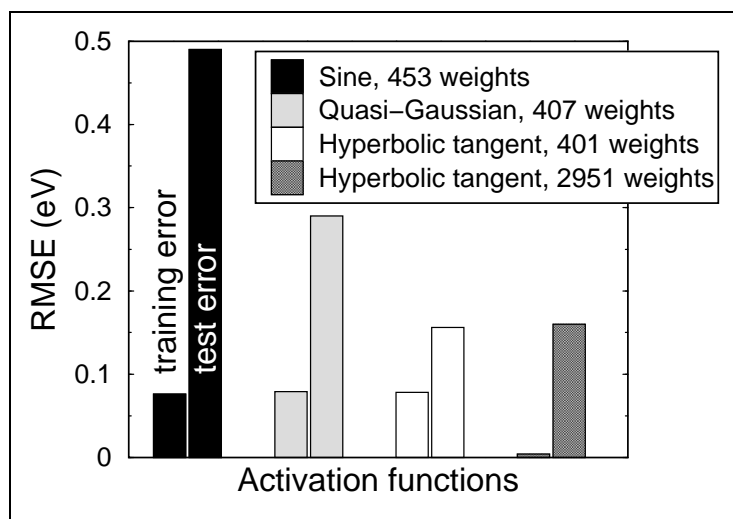
Figure 8.6: Root mean squared error on the training set versus the number of epochs with two different forgetting schedules according to  $\lambda(k) = \lambda_0 \lambda(k-1) + 1 - \lambda_0$ , where  $k$  is the number of updates. The cost function is  $\epsilon(k) = \sum_{p=1}^k \|e(p)\|_2 \lambda^{k-p}$ , where  $k$  is the number of updates and  $e(p)$  is the current error.

### 8.2.2 Activation functions

The flexibility of neural networks originates from the non-linear activation functions of its hidden layers. Sigmoidal functions like the hyperbolic tangent are the most common forms of transfer functions, but also sine-functions and more recently Gaussians have been applied. A general concept of approximation with neural networks is to choose one certain class of activation functions *a priori* and to stick to them. The fitting boils then down to find the appropriate number and values of parameters for an optimal solution. However, we wanted to test the different choices of activation functions for the approximation of potential-energy surfaces. As the training set we employed 1560 examples of the six-dimensional analytical PES. The test set consists of 7200 examples. The desired accuracy of our interpolations was a RMSE of 0.1 eV, which is the corresponding accuracy of density functional theory calculations.

Fig 8.7 displays the root mean squared error after 100 epochs for the different activation functions. For sine functions a number of 453 parameters was necessary to achieve a RMSE of the training set below the desired threshold of 0.1 eV. However, the value of the test set error of 0.49 reflects a poor generalisation ability of networks with these basis functions. This is clearly visible in the two elbow plots of the neural network PES in Fig. 8.8. A comparison with the underlying analytical PES in Fig. 8.1, p. 74, reveals a locally good, but globally not smooth enough approximation of a network with trigonometric activation functions. It fits the training set well, but produces a rather *bumpy* output function. A higher number of parameters decreases the training error further but also leads to even more pronounced overfitting.

Figure 8.7: Training and test root mean squared error for different activation functions. Training set: 1560 examples. Test set: 7200 examples. The number of parameters of the different networks is indicated in the legend. The number of epochs is 100.



With the explicit use of Gaussians as activation functions we were not able to obtain a good fit at all. Such radial basis function networks (RBFN) need a different initialisation scheme for the network weights, a clustering algorithm. With a random choice of the initial parameter set as employed in our program a minimisation to a satisfactory level of error for RBFN's is difficult to achieve. However, Flake proposed an easy trick to simulate Gaussian-like functions [139]. Let us consider the following functions:  $\text{sigmoid}(x) = 1/(1 + \exp(-x))$  and  $\text{gauss}(x) = \exp(-x^2)$ . We can define a

quasi-Gaussian as  $q(x) = 2 - 2/(gauss(x) + 1) = 2 - 2sigmoid(x^2)$ . This means that a gaussian-like function can be formed from an affine transformation of a sigmoid whose input has been squared. A similar transformation can be applied to a hyperbolic tangent. Hence, with a simultaneous presentation of  $x$  and  $x^2$  to the network, the architecture has the localised properties of an RBFN in addition to the usual global properties. In practice, the test error has indeed reduced to 0.29 eV for such a network with 407 weights (Fig. 8.7). However, this level of error is still not acceptable.

The sigmoidal functions are the most common forms of activation functions. The training error in Fig. 8.7 for a network with hyperbolic tangents and a parameter number comparable to the last examples, namely 401, reaches again values within the desired *ab initio* accuracy. More importantly, the test error also drops to 0.16 eV, the generalisation ability is significantly improved. A closer look at the PES in Fig. 8.9 demonstrates that the network output function with sigmoidal functions is very smooth. However, the network is not able to distinguish between the two adsorption configurations. Both plots exhibit a barrier to dissociation, but only the right elbow should have one, see Fig. 8.1, p. 74. The corresponding region is the most important part of the PES where the bond of the molecule breaks. A deviation of the NN-PES from the analytical PES in this region will have important dynamical consequences. Even with a higher weighting of the bond-breaking data points the network was not able to reproduce the correct co-existence of activated and non-activated paths. A different sampling of the points with a concentration on the most important parts of the PES did neither lead to a satisfactory result. The described neural network is not flexible enough to fit such a detailed PES. Even though the approximation error lies within the range of the *ab initio* accuracy, the PES is not represented correctly in its most crucial part. From this we deduce that it is not always sufficient to judge an optimal fit from the RMSE alone. This is a very important point and should be kept in mind. Since the training and test set will always result from a similar sampling of the problem, one has to be very careful in the interpretation of the RMSE.

If we increase the number of parameters to a value of nearly 3000, the training error as displayed in Fig. 8.7 drops by an order of magnitude to 0.004 eV without leading to an increase of the test error. The elbow plots in Fig. 8.10 give evidence of the global and local approximation ability of neural networks. Although a large number of parameters are utilised in the fit, the output function stays rather smooth and very importantly, distinguishes now between the two adsorption configurations. These results can be explained as follows. The dissociation PES consists of numerous local bumps in the bond-breaking region, one in each 2D cut of the 6D PES. In order to form a peak with sigmoidals many of them - rotated around the centre of the hill - are necessary. Consequently, in order to properly describe the process of bond-breaking within a very localised region of a detailed PES and at the same time modelling a smooth function outside that region, a large number of Fermi-like basis functions is required.

Furthermore, we found that a detailed PES can be fitted with less complexity if a network with two hidden layers is chosen as presented above. In  $N$  dimensions  $2N$  nodes in the first hidden layer and one node in the second hidden layer can form one bump [52]. Of course, such a function can also be approximated to arbitrary accuracy with only one hidden layer, but in this case even more units have to be employed.

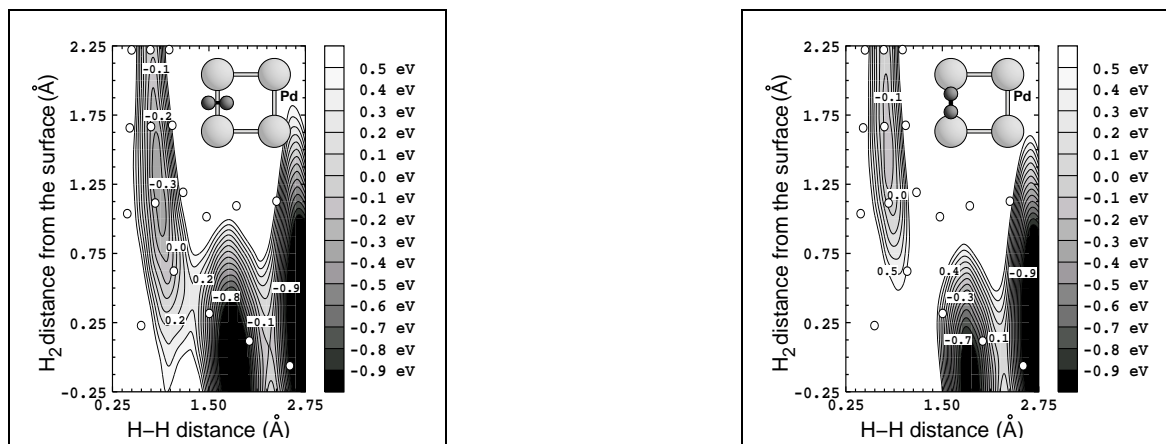


Figure 8.8: Neural Network fit with **sine** functions as activation functions in hidden layer. The fit is performed with **453** parameters and a 6–45–1 net.

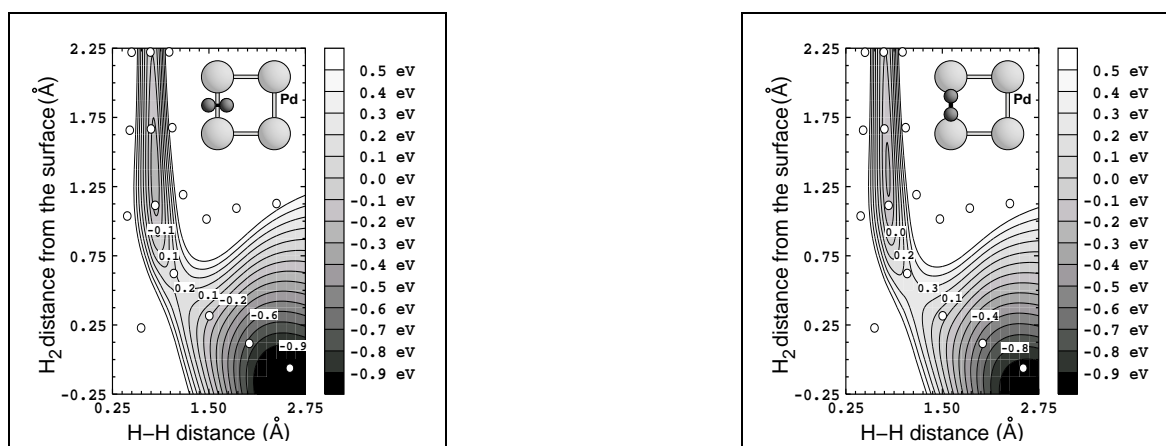


Figure 8.9: Neural Network fit with **hyperbolic tangents** as hidden layer activation functions. The fit is performed with **401** parameters and a 6–16–16–1 *tl* net.

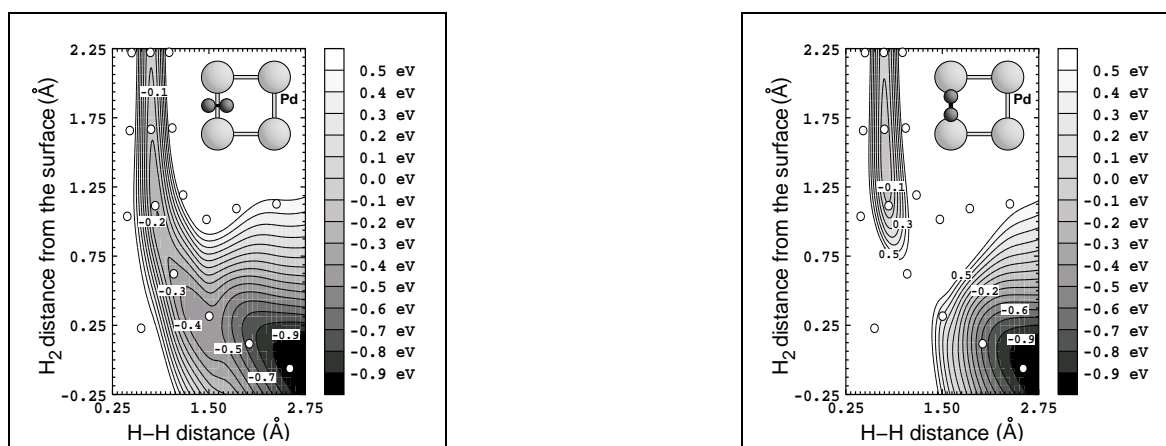


Figure 8.10: Neural Network fit with **hyperbolic tangents** as hidden layer activation functions. The fit is performed with **2951** parameters and a 6–50–50–1 *tl* net.

### 8.2.3 Ill-conditioning

Neural networks approximation problems are - as explained in Sec. 5.11, p. 58 - nearly always ill-conditioned. We will now discuss a trick to reduce this problem for gradient-based learning methods by v. d. Smagt and Hirzinger [134].

One reason for ill-conditioning stems from the structure of neural networks itself. Due to the chain rule for calculating the Jacobian, the gradients in the lower-layer weights are related to the higher-layer weights. If a node has a very small outgoing weight then the influence of the incoming weights to that unit will be diminished. This results in flat spots in the error surface, which translates into slow training. The solution is to add linear shortcut connections from the input to the output nodes to create the *linearly augmented feed-forward network*. These connections share the weights with the input to hidden connections so that no new weights are added as displayed in Fig. 8.11. This trick enhances the sensitivity of the network to those incoming weights thus reducing the flat spots in the error surface. The condition number of the Hessian approximated by the squared Jacobian can be many orders of magnitude smaller with a network inhibiting shortcuts, see Fig. 8.12. In the presented example for a 6–30–1 *tl* net and the same data set as before, the condition number without shortcuts reaches a value of  $10^{16}$ , which results in numerical instabilities. The test error with direct links reaches a value of 0.21 eV compared to 0.3 eV for a network without shortcuts. However, the training does not often lead to such high condition numbers. Only in rare cases direct links improve the training results.

Within the EKF algorithm the growth and shrinkage of the covariance matrix, which is an approximation of the Hessian, can be controlled by the parameters of the weighting scheme [127]. An unwilling growth of the condition number of the approximated Hessian can also be suppressed by choosing smaller forgetting schedule parameters. With this approach, we were able to improve the results more efficiently than with the application of direct links.

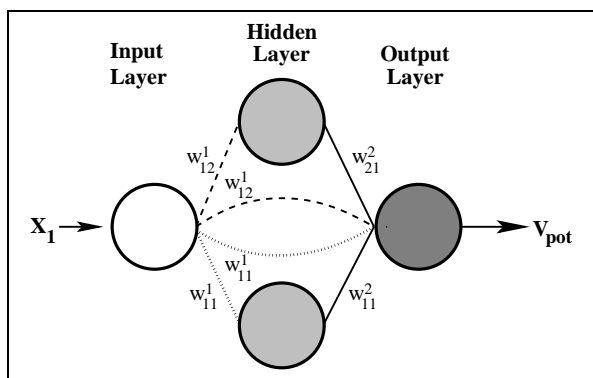


Figure 8.11: *Linearly augmented feed-forward network with shortcuts from the input layer to the output layer. The direct links from the inputs share the weights with the connections to the hidden layer.*

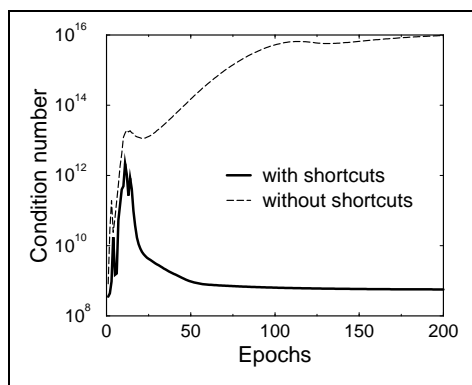


Figure 8.12: *Condition number of the Hessian approximated by the square Jacobian as function of the epochs for a neural network with shortcuts and without.*

### 8.2.4 Incorporation of the symmetry

The computational effort in DFT calculations can be enormously reduced by taking advantage of the symmetry of the underlying problem. With the simple two dimensional trigonometric function in Sec. 7.3 we demonstrated the ability of neural networks to *learn* symmetry. Yet, since we know the surface symmetry beforehand it will be advantageous to include this knowledge into the approximation prior to the minimisation. In this way we let the network concentrate on the crucial process, the bond-breaking of the molecule. In order to do so we pre-process the coordinates of the problem.

The original set of coordinates  $X_c, Y_c, Z_c, d, \theta, \phi$  describe the six degrees of freedom of the molecule. Due to the high costs of *ab initio* calculations information on the clean Pd(100) has been determined only on the edges of the irreducible part of the unit cell, the filled area in Fig. 8.13 [29, 22]. In order to represent the whole surface area the analytical fit assumed a certain set of symmetry operations to be valid [41]. We point out that the applied symmetry introduces artificial features into the PES. For instance, the molecule in the analytical PES does not have any  $\phi$  dependency on the diagonals of the unit cell. However, this affects only a small part of the configuration space. Furthermore, the analytical PES has already proven to represent the dissociation process correctly [41]. Since this PES serves as a test problem for our neural network approach, we employed the same symmetry and transformed the original coordinates into a set of eight inputs to the neural network:

$$X_1 = d,$$

$$X_2 = d^2,$$

$$X_3 = Z_c,$$

$$X_4 = \sin^2(\theta) \cos(2\phi) [\cos(G \cdot X_c) - \cos(G \cdot Y_c)],$$

$$X_5 = \sin^2(\theta) \cos(2\phi) [\cos(2G \cdot X_c) - \cos(2G \cdot Y_c)],$$

$$X_6 = \cos^2(\theta),$$

$$X_7 = \cos(G \cdot X_c) + \cos(G \cdot Y_c),$$

$$X_8 = \cos(2G \cdot X_c) + \cos(2G \cdot Y_c).$$

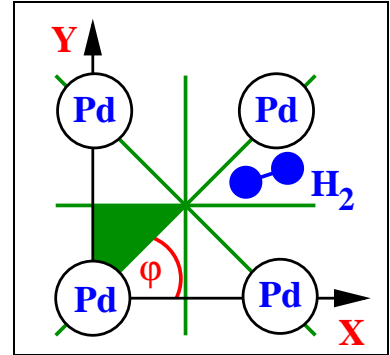


Figure 8.13: Surface unit cell and coordinates of the system  $H_2/Pd(100)$ . The filled area marks the irreducible part of the unit cell.

The transformations are based on Fourier terms in the lateral coordinates  $X_c$  and  $Y_c$  up to a reciprocal lattice vector of  $2G$  representing the periodicity of the surface, with  $G = 2\pi/a$  and the lattice constant “ $a$ ”. The term  $\cos(2\phi) [\cos(G \cdot X_c) - \cos(G \cdot Y_c)]$  in the fourth coordinate reflects the four-fold symmetry of the surface. The factor  $\sin^2(\theta)$  weights this term, since the energy of an upright molecule should not have any azimuthal dependency. It also reflects the internal symmetry of the diatomic molecule. From the theoretical *ab initio* calculations it has been found that the energy increases like  $\cos^2(\theta)$  [29], which we included as one input. There is no symmetry within the

coordinates  $d$  and  $Z_c$ . However, the vibration of the molecule in the gas phase can be described by a harmonic oscillator and therefore we incorporated an additional coordinate  $d^2$ .

Instead of presenting the original six degrees of freedom of the molecule to the neural network we now apply this new set of eight inputs representing the symmetry of the surface. The neural network performs a *non-linear fit* on these new inputs. The transformation needs to be done only once per surface symmetry.

### 8.2.5 Optimised neural network structure

In summary, our neural network approximations to potential-energy surfaces will be based on several features. As an optimisation algorithm of the network weights we will employ the adaptive global extended Kalman filter (AGEKF) with two forgetting schedule parameters  $\lambda(0)$ ,  $\lambda_0$  and an adaptive threshold of  $a_{th} \times RMSE$ .

The activation functions of the hidden layers are hyperbolic tangents and linear functions in the output layer. The neural network structure will mainly consist of one input layer, two hidden layers and one output layer with a high number of parameters.

The input data are pre-conditioned as described in the theoretical part, i.e. we subtract the means and normalise the variances. Furthermore, we pre-process the data and transform the inputs to include the surface symmetry.

In order to ensure a most accurate representation of the potential we use individual weighting of each energy. For instance, the dissociation dynamics depend crucially on the region in which the bond of the molecule breaks, whereas the part where the potential is already elevated is of less importance. We will associate the former region with weights which are up to ten times higher than the rest of the geometries.

## 8.3 Neural Network PES

We will now present six dimensional interpolations of the *analytical* potential-energy surface for the dissociation of hydrogen over Pd(100) with neural networks. Open questions are the necessary number of training points and their sampling for obtaining a good description of the PES.

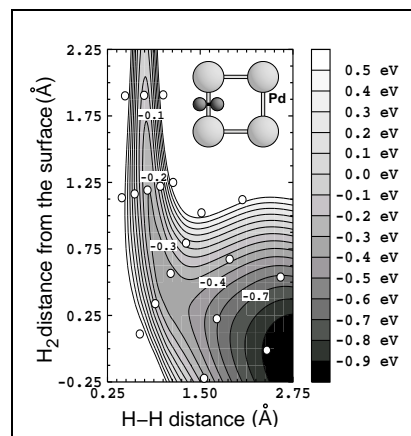
The usual approach in theoretical *ab initio* studies of dissociation processes is based on the calculation of 2D sections of the 6D energy surface, the elbow plots. For one such section, the orientation of the molecule ( $\theta$  and  $\phi$ ) and the coordinates of the centre of mass in the surface plane ( $X_c, Y_c$ ) are kept fixed. Only the height  $Z_c$  and the bond length  $d$  vary. Commonly these elbow plots are evaluated with the molecule above high-symmetric sites as discussed in detail in Sec. 3.3, p. 33. We will adopt this approach here as well and sample the points from the analytical PES in the same way. The neural network is then used to interpolate between these sections.

The accuracy of neural network models is usually measured by the root mean squared error (RMSE) alone. As an additional check of the accuracy we will compare the sticking probability for the analytical and the neural network PES determined by classical molecular dynamics calculations.



### 8.3.1 Neural Network fit of a 2D elbow plot

As a first test of the neural network approach to fitting an analytical PES we approximated a two-dimensional elbow plot, namely the minimum pathway for the dissociating hydrogen molecule. In the corresponding configuration the molecule approaches the substrate over the bridge site with its molecular axis parallel to the surface. We obtained a good interpolation with 31 training points sampled along and perpendicular to the reaction path as shown in Fig. 8.14. The white circles indicate the positions of the training points obtained from the analytical PES, cf. the analytical cut in Fig. 8.1(a), p. 74. The interpolation has been performed with a 1–7–7–1 *tl* network with 29 parameters. Since the analytical PES is formulated in reaction path coordinates, we were able to weight the data points along the reaction path differently (10 times higher) than the other examples. The training error after 1000 epochs measured 0.044 eV and the test error calculated with a set of 150 points was 0.053 eV and therefore within the range of *ab initio* accuracy.



**Figure 8.14:** 2D neural network fit to the analytical elbow plot of Fig. 8.1(a).

### 8.3.2 NN-fit based on high-symmetric configurations

We trained a 8–50–50–1 *tl* neural network with 1560 examples calculated from the analytical PES. The elbow plots were evaluated above different high-symmetric sites, i.e. top, bridge and hollow sites and one intermediate configuration at ( $X_c = 0.25a, Y_c = 0.25a$ ,  $a$ : lattice constant). At each site the energies were collected for five different angles  $\phi$  with the molecule upright, 45° tilted and parallel to the surface. For a single elbow plot we used 30 points along and perpendicular to the reaction path. The test set consists of 5200 energies sampled from the same elbow plots as the training set. The training error after 50 epochs and two hours runtime on an IBM-SP2 node measured 0.1 meV with a test error of 0.15 eV. From the information of the root mean squared error alone we would judge this approximation as being satisfactory.

The obtained neural network served as an input to classical molecular dynamics calculations. Fig. 8.15 illustrates the adsorption process over one particular site, the bridge site with the hydrogen atoms pointing towards the on-top sites. The molecule approaching the surface under normal incidence with its axis parallel to the surface at an energy of 0.5 eV is not able to overcome the barrier for dissociation. Due to the highly repulsive palladium top sites it is scattered back into the gas phase (Fig. 8.15(a)). With a kinetic energy of 0.9 eV the molecule has enough momentum to overcome the energy barrier and dissociates (Fig. 8.15(b)).

From the molecular dynamics simulations we calculated the sticking probability of the impinging hydrogen molecule as a function of the initial kinetic energy. The dissociation process is highly site dependent which requires to consider a good statistical average over the initial configurations for the determination of the sticking coefficient.

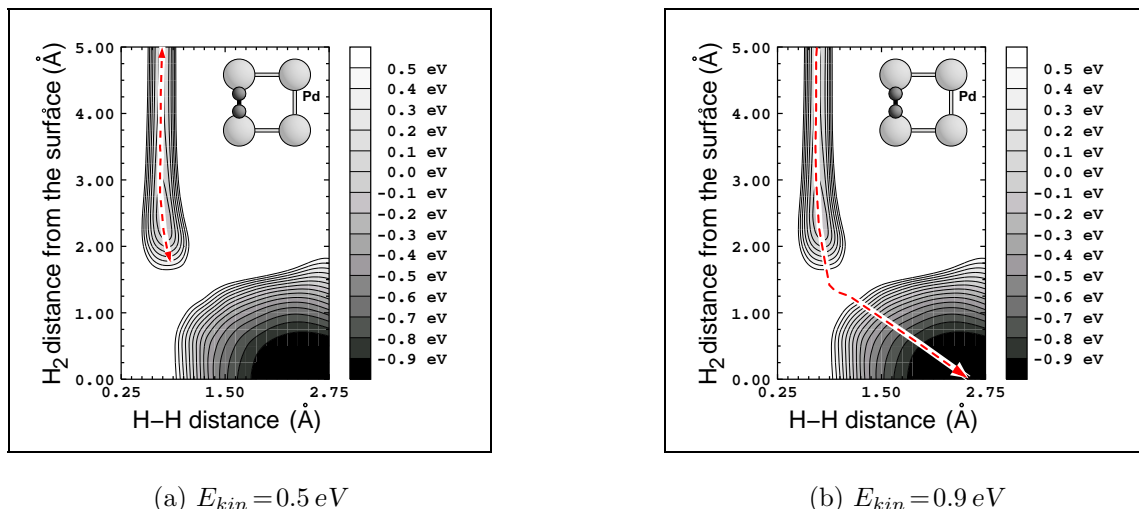


Figure 8.15: Two classical molecular dynamics trajectories (dashed lines) on a NN-PES. The initial conditions of the molecules are the same except for their kinetic energy. The simulation time was (a) 52 fs and (b) 40 fs. Insets: configuration of the molecule.

For each kinetic energy we need to calculate 500–1000 trajectories with random sampled initial configurations until convergence of sticking is attained. The error of the sticking coefficient is corresponding to  $1/\sqrt{n}$ , where  $n$  is the number of trajectories. For each sticking curve the sticking probability has to be evaluated at a number of energies depending on the energy range of interest. For the presented adsorption coefficients we performed MD calculations with 10000–30000 trajectories.

Fig. 8.16 compares the sticking probability obtained from the neural network PES with the dynamical result from the underlying analytical PES. The neural network PES interpolating high-symmetric sites reproduces the increase of the sticking probability at energies larger than 0.2 eV qualitatively, but it fails to reproduce the high sticking probability at low kinetic energies. It has been shown that this feature of the

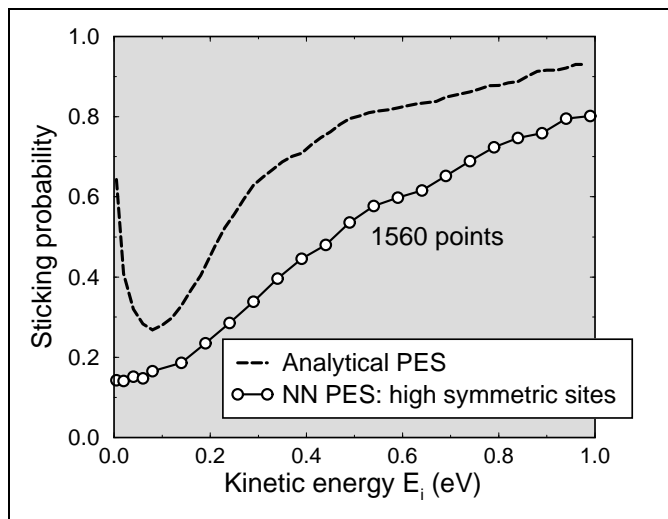


Figure 8.16: Sticking probability versus kinetic energy for the system  $H_2/Pd(100)$ . The sticking has been calculated by classical molecular dynamics on a six-dimensional analytical and neural PES, respectively. Training set: 1560 examples.

dissociation process is due to the steering effect. At low energies the forces can reorient the molecule towards the minimum path for dissociation. At higher energies steering is less effective leading to a drop of the sticking probability.

Fig. 8.17(b) and Fig. 8.17(c) show the corrugation of the barrier heights calculated from the analytical and neural PES, respectively. The plots have been obtained by fixing the hydrogen molecule at a height of  $z=1.6 \text{ \AA}$  above the surface with an intramolecular distance of  $r=1.0 \text{ \AA}$  and angles  $\phi=\pi/2$ ,  $\theta=\pi/2$  while changing the lateral coordinates across the unit cell as illustrated in Fig. 8.17(a). The configuration of the molecule corresponds to the region where the bond already started to break.

In the analytical corrugation of Fig 8.17(b) there is a high barrier for dissociation present if the molecule approaches the surface above the top site. Above the bridge and the hollow site the molecule is able to dissociate freely. Furthermore, the energy barrier decreases monotonically from the top site to the bridge site. A slow molecule is able to move from the top site where it experiences a high barrier to the favourable dissociation configuration above the bridge site. It is also able to reach the bridge site from the hollow site. This is not true for the neural network PES interpolating the top, bridge, hollow and one intermediate site as illustrated in Fig 8.17(c). The PES exhibits a small barrier of 0.1 eV between bridge and top site and a large barrier of 1.0 eV between bridge and hollow site. These artificial barriers diminish the steering effect and thus cause a monotonically increasing sticking curve as shown in Fig. 8.16.

We conclude that for interpolations of potential-energy surfaces with neural networks it is essential to include more than the usual calculated elbow plots above high-symmetric sites in the training and test sets. For instance, if we apply additional configurations in the *test set* of the above presented NN-approximation we get a test error of 0.32 eV, which is clearly above the desired accuracy. Hence, with the use of additional configurations also the RMSE reflects the unsatisfactory interpolation based on high-symmetric sites only.

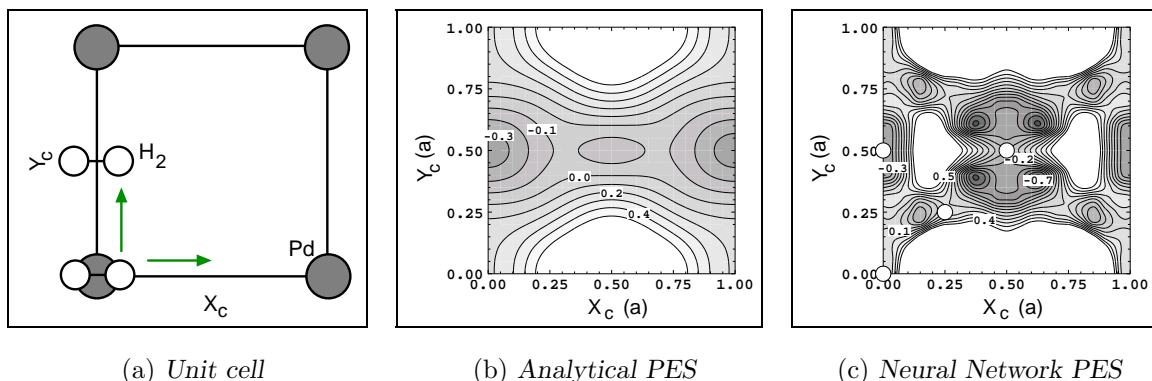
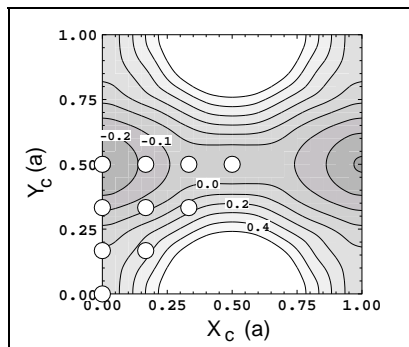


Figure 8.17: Corrugation of the energy across the unit cell. (a) Schematic drawing of the configurations used for scanning the barriers. The molecule is oriented parallel to the surface with its bond axis parallel to  $X_c$  at a height  $z=1.6 \text{ \AA}$  with a bond length  $d=1.0 \text{ \AA}$ . (b) and (c): Potential energies as a function of the lateral coordinates  $X_c$  and  $Y_c$ . The contour spacing is 0.1 eV. The circles in (c) mark the NN training points.

### 8.3.3 NN-fit based on an enhanced lateral grid

In order to achieve a better representation of the steering effect with neural networks we increased the number of training points in the lateral directions of the unit cell. Instead of applying only four lateral configurations we used ten different adsorption sites in the



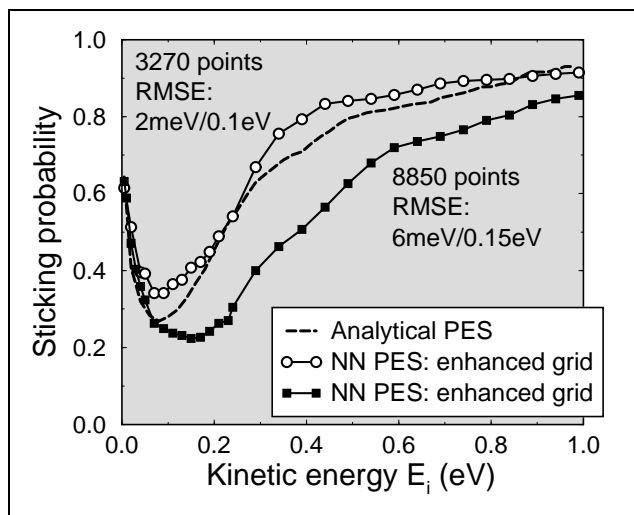
**Figure 8.18:** *Corrugation of the Neural Network PES based on an enhanced lateral grid.*

irreducible part of the unit cell. The lateral mesh is indicated by the white circles in Fig. 8.18 where we plotted the corrugation of the energy barriers for a NN-PES based on the enhanced mesh. The energies have been obtained in the same way as described in Fig. 8.17. The corrugation is now well represented, cf. the analytical corrugation in Fig. 8.17(b).

We performed a number of interpolations of the analytical PES with different training sets using a  $9-50-50-1$  *tl* neural network. Fig. 8.19 displays the dynamical results of two of them. For the interpolation with 3270 training points with the above introduced enhanced lateral grid - while keeping the sampling of the other dimensions as described in the previous section - the sticking probability of the analytical PES is well reproduced. The training and test error after 20 epochs were 2 meV and 0.1 eV.

In Fig. 8.19 we have also plotted the result obtained from a less good neural network fit. The training and test errors with 6 meV and 0.15 eV based on a training set of 8850 energies were slightly worse. The higher training and test errors lead to a larger deviation of the sticking probability from the analytical PES. We point out that it may always be possible that a better neural network fit with a different set of Kalman filter parameters and a different number of weights exists. Yet, it will be difficult to judge an optimal interpolation without knowing the dynamical result *a priori*.

We emphasise that the NN-PES with this finer lateral grid is not uniquely defined. An increase of the number of points in the other degrees of freedom as done for the training set with 8850 energies, does not necessarily lead to a better fit. All degrees of



**Figure 8.19:** *Sticking probability versus kinetic energy for the system  $H_2/Pd(100)$  for two different training sets based on ten lateral configurations of the molecule. The training and test errors for each fit are indicated in the graph. Higher errors lead to a less good sticking curve. Neural Network:  $8-50-50-1$  *tl*. Extended Kalman Filter parameter:  $\lambda(0)=0.98$ ,  $\lambda_0=0.99936$ ,  $a_{th}=0.6$ .*

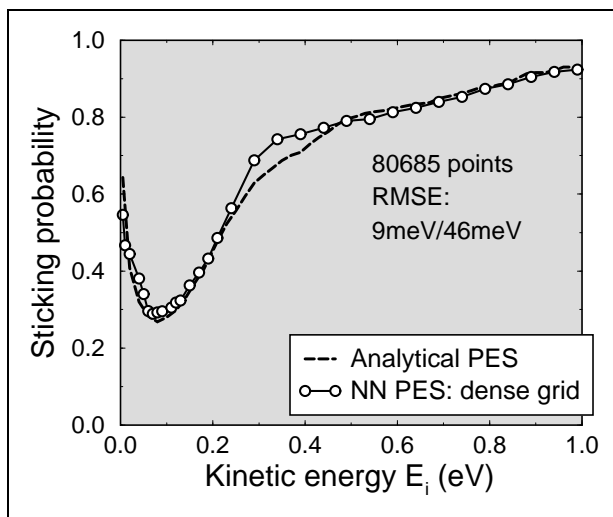
freedom play a role and the mesh is not dense enough to reflect that a higher number of points should lead to a quantitatively better dynamical result. In any case, the NN-models based on an enhanced lateral mesh reproduced the steering effect in all cases qualitatively.

### 8.3.4 NN-fit based on a dense grid of configurations

In order to get also quantitative agreement with the underlying analytical PES we further increased the density of the mesh for the data sampling. We will now present the results of a neural network fit based on a dense grid of potential energies taken from the analytical potential-energy surface. The corresponding training set is by an order of magnitude larger than before and consists of 80685 energies evaluated above 55 adsorption sites. For the test set we collected 91665 points. After 20 epochs the training error measured 9 meV with a test error of 46 meV. Both errors lie well below the desired *ab initio* accuracy of 0.1 eV.

With this dense grid in all six degrees of freedom of the hydrogen molecule we were able to get perfect agreement between the analytical and neural sticking curve as displayed in Fig. 8.20. Both the initial high adsorption probability followed by a drop of sticking due to the steering effect and the increase with higher kinetic energies typical for dissociative adsorption are well reproduced. The differences between the analytical and neural sticking curve are smaller than 5% over the presented energy range. The results demonstrate that the steering effect involves all six degrees of freedom of the test problem and underline the importance of high-dimensional studies in order to predict reaction probabilities.

Figure 8.20: Sticking probability versus kinetic energy for the dissociation of hydrogen on the Pd(100) surface. The training set consists of 80685 training points sampled on a dense grid in all six dimensions, with e.g. 55 lateral configurations. The fit is based on a 8–50–50–1 *tl* neural network. Parameter set:  $\lambda(0)=0.98$ ,  $\lambda_0=0.99936$ ,  $a_{th}=0.6$ .



Due to the numerical costs the mapping of potential-energy surfaces with density functional theory methods is currently limited to a number of the order of  $10^2 - 10^3$  energies. Direct molecular dynamics simulations of dissociation reactions on surfaces, where the potential energy and the forces are being evaluated *on the fly*, can thus require to calculate a number of the order of  $10^7$  total energies. We have shown that

an interpolation with neural networks of the discussed detailed PES requires to consider a number of  $10^4$ – $10^5$  points. The costs for a description of such reactions with neural networks are therefore orders of magnitude smaller compared to direct *ab initio* dynamics.

Fig. 8.21 illustrates the accuracy of the obtained neural network model. It shows the distribution of prediction errors for the test set. 94% of the test examples have an error smaller than 0.1 eV and already 99% do not exceed a threshold of 0.2 eV. The distribution of errors as a function of the potential energy in Fig. 8.22 reveals that larger errors occur only at values above 1 eV. This is the region far away from the valley of the elbow plots. The errors were influenced by the imposed higher weighting of the points close to the minimum dissociation pathway. However, the results support that indeed the regions of higher potential energies have almost no influence on the reaction probabilities as plotted in Fig. 8.20.

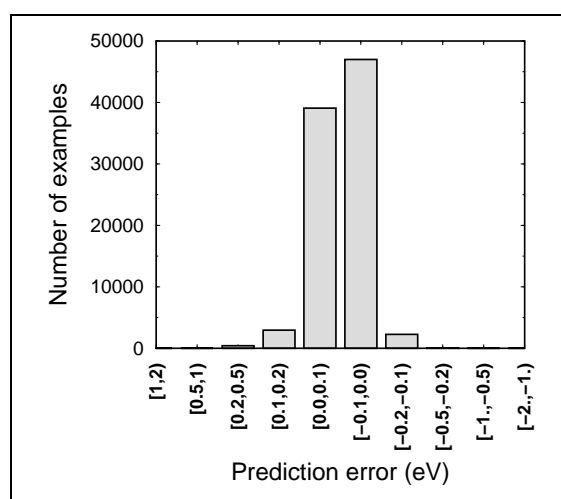


Figure 8.21: Number of test examples versus prediction error for the training with a dense mesh of energies.

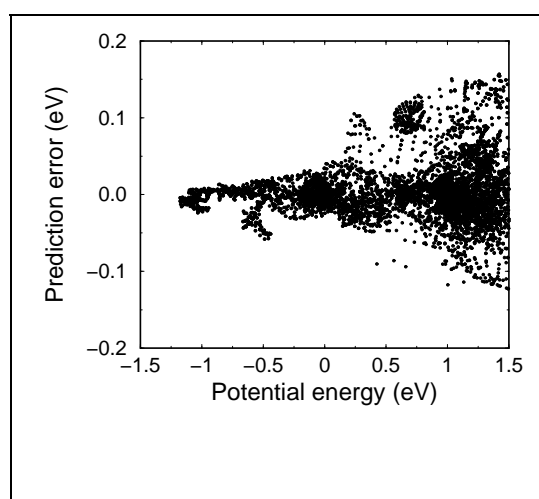


Figure 8.22: Prediction error of the test set versus potential energy for the training with a dense mesh of energies.

This is an important issue for the fitting of potential-energy surfaces. Not all configurations are equally important for the determination of the sticking probability. For instance, at lower kinetic energies the adsorption dynamics depend crucially on whether there is a small energy barrier in the entrance channel, where the centre of mass of the molecule is still far away from the surface, or not. Yet, the region where the potential is already elevated might have almost no influence on the dissociation probability. Consequently, the root mean squared error, which is usually used as a measurement of the accuracy of the fit, is of less significance. The developed neural network program allows individual weighting of each input energy. We will therefore always weight the configurations close to the valley of each dissociation pathway up to 10 times higher than the rest of the geometries.

---

## Chapter 9

### Neural Network test: 6-D analytical PES for $\text{H}_2/(2 \times 2)\text{S}/\text{Pd}(100)$

---

In the previous chapter we discussed a system in which *activated and non-activated* pathways towards dissociation existed on the same surface, with the former ones being a minority but having important dynamic consequences. We showed that in order to obtain a very good agreement between the analytical and neural sticking probability a high number of training points and parameters were required. Still, in comparison to *direct ab initio molecular dynamics* orders of magnitude fewer density functional theory calculations were necessary to gain reliable dynamical properties.

As a second test problem for the interpolation of the PES in dissociation reactions with neural networks we investigated a system with *activated* paths only: The dissociation of  $\text{H}_2$  over a sulphur covered  $\text{Pd}(100)$  surface. It is experimentally well known that sulphur adsorbates hinder the  $\text{H}_2$  dissociation process on  $\text{Pd}(100)$  [4]. This observation was verified by density functional theory studies [29, 30, 33]. It was found that the reaction is still exothermic. Thus, the poisoning effect of sulphur adatoms for  $\text{H}_2$  dissociation at low sulphur coverages ( $\Theta_S \leq 0.25$ ) is governed by the formation of energy barriers and not by blocking of adsorption sites.

A study of an activated system with neural networks will be interesting for various reasons. First of all, a PES with energy barriers in every adsorption geometry may be easier to interpolate instead of fitting a coexistence of pathways with and without barriers. Furthermore, we are interested in the sticking probability of the system  $\text{H}_2/\text{Pd}(100)$  with potassium adsorbates. Therefore it will be ideal to test the fitting for the same surface with a different adsorbate, i.e. sulphur. Moreover, it has been shown by *ab initio* calculations that the dissociation on the potassium covered and on the sulphur covered  $\text{Pd}(100)$  are both activated processes [29, 37].

For the sampling of the data from the analytical PES we will follow the common scheme of calculating two dimensional cuts in the  $Zd$  coordinates while keeping the other coordinates fixed. We will then interpolate these cuts with neural networks. However, we will also present a modified sampling scheme for dissociative reactions which allows to obtain reliable dynamical results with a small number of energies. Furthermore, we will present a constrained fitting approach with neural networks tailored to the specific needs of dissociative reaction dynamics.

## 9.1 *Ab initio* and analytical PES

Density functional theory calculations of the system  $\text{H}_2/(2 \times 2)\text{S}/\text{Pd}(100)$  revealed that the PES is changed significantly compared to the dissociation on the clean  $\text{Pd}(100)$  surface [29,30,33]. While the process on the latter surface is non-activated, for a  $(2 \times 2)$  sulphur adlayer corresponding to a coverage of  $\Theta_S = 0.25$  it is inhibited by energy barriers. Their heights depend strongly on the distance between the hydrogen and the sulphur atoms leading to a highly corrugated PES. Due to the strong repulsion between sulphur and hydrogen the largest barriers are in the vicinity of the sulphur atoms. The minimum barrier towards dissociative adsorption has a height of 0.1 eV, while close to the adsorbate atoms the barriers become larger than 2.5 eV. The adsorption height of the sulphur atoms is 1.31 Å above the surface. The adsorption energy at all sites close to sulphur atoms is reduced in comparison to the clean surface. But still,  $\text{H}_2$  adsorption into all hollow sites not occupied by sulphur remains an exothermic process.

For the theoretical investigation of the high-dimensional PES the common strategy of the computation of 2D cuts through the 6D configuration space has been followed. These cuts have been interpolated analytically. The representation is based upon the analytical PES for the clean  $\text{Pd}(100)$  surface [33]. Due to the larger unit cell some higher Fourier coefficients have been included in the lateral directions. Also in the azimuthal dependence a higher order term has been introduced. Again, the coordinates in the  $Zd$  plane have been transformed into reaction path coordinates. The parameters of the interpolation have been determined such that the difference to the *ab initio* calculations on the average is smaller than 50 meV.

Fig. 9.1 shows two 2D-cuts through the six-dimensional configuration space, compare with the corresponding *ab initio* cuts in Fig. 2.8, p. 23. Whereas on the clean surface the molecule over the palladium bridge site was able to dissociate freely, due to the presence of sulphur the molecule experiences a barrier of 0.16 eV. The minimum pathway is now over the fourfold hollow site with an energy barrier of 0.11 eV.

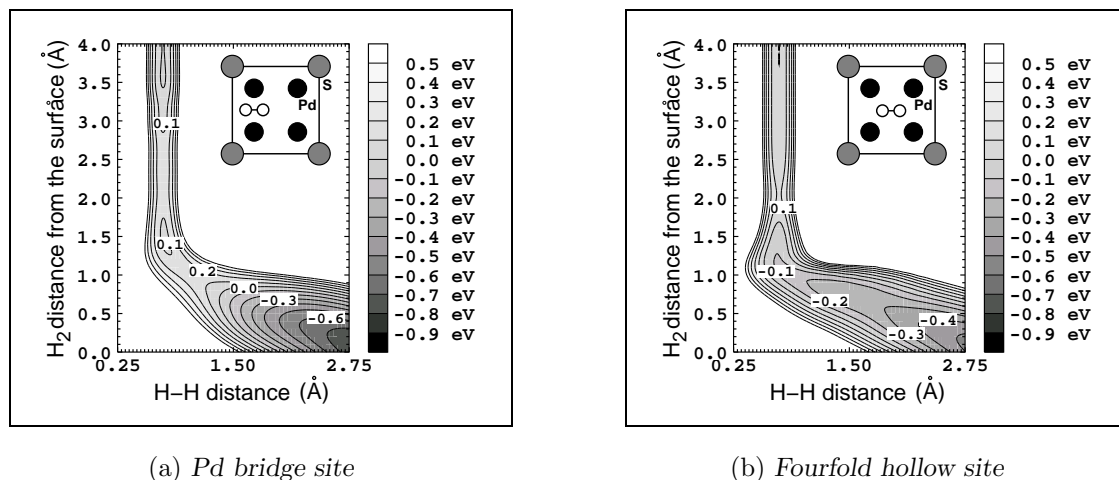


Figure 9.1: Contour plots through the six-dimensional analytical PES of the dissociation of  $\text{H}_2$  over  $(2 \times 2)\text{S}/\text{Pd}(100)$ . Insets: geometry of the dissociation pathways.



## 9.2 Incorporation of the symmetry

In order to *a priori* represent the symmetry within the neural network we used the same terms as on the clean Pd(100) surface but added one higher order term for the azimuthal dependency. In analogy to the analytical PES we employed reaction path coordinates in the  $Zd$  plane. Furthermore, we did not employ the distance of the hydrogen molecule from the surface as an input to the neural network, but rather an exponential decay of that coordinate. In reaction path coordinates this translated to the term  $e^{(-s/2)}$ , where  $s$  is the coordinate along the reaction path. The transformation reflects that far away from the surface the molecule is in the gas phase and any dependency on the distance from the substrate should vanish. Moreover, in the gas phase the potential energy is isotropic. Only the bond length of the two hydrogen atoms should play a role, and therefore we weighted all other terms with the same factor  $e^{(-s/2)}$ .

However, we point out that these transformations are not a requirement for fitting PESs with neural networks. The network will be able to learn the desired dependency during minimisation without explicit information. But it is always advantageous to use all the physical knowledge available and to let the neural network concentrate on the unknown part of the problem. The new set of nine coordinates, i.e. the inputs to the neural network, are:

$$X_1 = d,$$

$$X_2 = d^2,$$

$$X_3 = e^{(-s/2)},$$

$$X_4 = \sin^2(\theta) \cos(2\phi) [ \cos(G \cdot X_c) - \cos(G \cdot Y_c) ] e^{(-s/2)},$$

$$X_5 = \sin^2(\theta) \cos(2\phi) [ \cos(2G \cdot X_c) - \cos(2G \cdot Y_c) ] e^{(-s/2)},$$

$$X_6 = \cos^2(\theta) e^{(-s/2)},$$

$$X_7 = [ \cos(G \cdot X_c) + \cos(G \cdot Y_c) ] e^{(-s/2)},$$

$$X_8 = [ \cos(2G \cdot X_c) + \cos(2G \cdot Y_c) ] e^{(-s/2)},$$

$$X_9 = \sin^4(\theta) \cos(4\phi) [ \cos(2G \cdot X_c) + \cos(2G \cdot Y_c) ] e^{(-s/2)}.$$

## 9.3 Neural Network PES

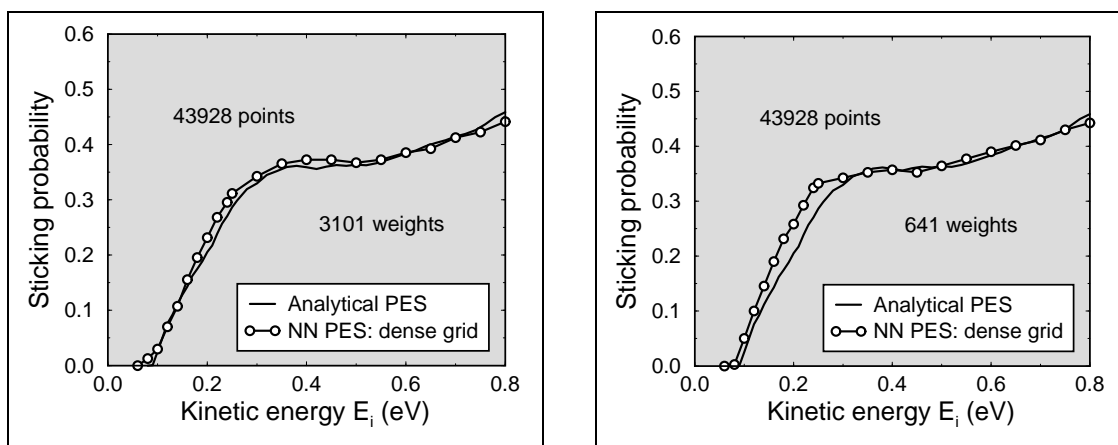
On the clean Pd(100) surface it was necessary to use a high number of training points along with a high number of parameters to represent the detailed potential-energy surface with activated and non-activated paths towards dissociation. Correspondingly, the first test of a neural network approximation of the analytical PES for the sulphur covered Pd(100) will be based on a dense grid of points.

### 9.3.1 NN-fit based on a dense grid of configurations

We fitted a number of 43928 examples from the analytical PES on a dense grid of configurations in all six degrees of freedom of the hydrogen molecule. The network consists of two hidden layers with fifty nodes in each of them (9–50–50–1  $tl$ ). For the test of the accuracy of the interpolation we used 5891 energies. After 40 epochs the training and test error were 0.033 eV and 0.043 eV. The neural network PES has subsequently been used in classical molecular dynamics calculations to determine the sticking probability. Fig. 9.2(a) displays these results. The neural network sticking curve agrees very well with the analytical sticking coefficient, their values differ by less than 3%. In comparison, for a good fit on the clean surface a number of examples twice as large was required.

Furthermore, for the sulphur poisoned surface the number of weights in the approximation can be greatly reduced without losing much of the networks performance. The sticking probability in Fig. 9.2(b) for a 9–20–20–1  $tl$  network differs from the value based on the analytical PES by less than 5%. The training and test error (0.068 eV and 0.081 eV) were slightly higher than for the network with 3101 parameters, but still within the desired *ab initio* accuracy. The training time with such a high number of examples but only 641 weights reduces to seven hours on an IBM-SP2 node in comparison to several days for the 3101 parameter case.

On the clean surface a neural network with such a small number of parameters was not able to describe the correct coexistence of activated and non-activated pathways. We conclude, with respect to the number of training points and the complexity of the appropriate neural network, that fitting a strictly activated potential-energy surface is a profoundly easier task.



(a) 9–50–50–1  $tl$  Neural Network

(b) 9–20–20–1  $tl$  Neural Network

Figure 9.2: Sticking probability versus kinetic energy for  $\text{H}_2/(2 \times 2)\text{S}/\text{Pd}(100)$  for two different Neural Networks. The data sampling from the analytical PES is based on a dense mesh of configurations in all six degrees of freedom of the  $\text{H}_2$  molecule. Parameter:  $\lambda(0) = 0.98$ ,  $\lambda_0 = 0.99906$ ,  $a_{th} = 0.6$ .

### 9.3.2 NN-fit based on eleven elbow plots

Currently, a dense grid of energies as presented in the previous section will not be available due to the high numerical costs of *ab initio* calculations. Commonly, density functional theory studies of PESs concentrate on two-dimensional cuts through the configuration space with the molecule above high-symmetric sites. We will therefore discuss how reliable reaction probabilities with neural network based on such two-dimensional cuts are.

In Fig 9.3 we plotted eleven such configurations of the system  $H_2/(2 \times 2)S/Pd(100)$ . The molecule approaches the surface above the fourfold hollow site, the palladium bridge site, the sulphur bridge site, on-top of a palladium atom and on-top of a sulphur atom. The orientation of the molecule is either parallel or perpendicular to the surface. These high-symmetric positions should represent the major features of the PES.

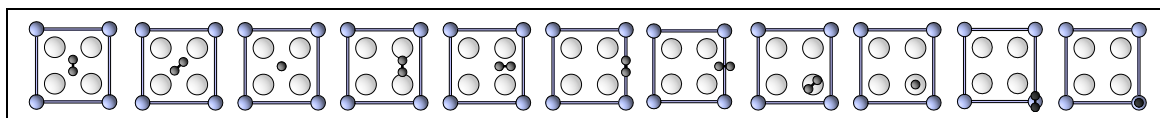
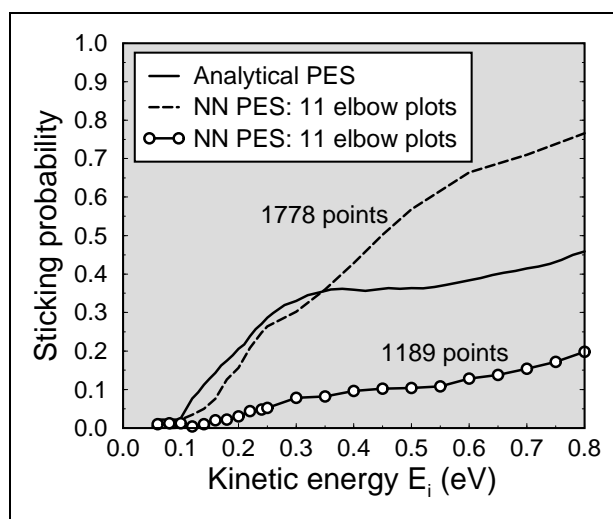


Figure 9.3: Eleven adsorption configurations of the system  $H_2/(2 \times 2)S/Pd(100)$ .

The NN interpolation based on a dense grid of points revealed that a network even with 650 parameters is able to determine the sticking curve of an activated system accurately. Though the performance of the neural network with a very large number of weights was slightly superior, such a network will also be prone to overfitting. We performed a 9–20–20–1 *tl* neural network interpolation based on 1189 training and 471 test energies obtained from the analytical PES in the configurations of Fig. 9.3. The test and training error after 100 epochs measured 0.078 eV and 0.096 eV, respectively. The resulting neural sticking coefficient in Fig. 9.4 exhibits the same increase of sticking with kinetic energy as the corresponding analytical curve but its value is strongly reduced. A neural network fit based on 1778 training examples from the same cuts resulted in a description of the PES which was too reactive at high kinetic energies (see Fig. 9.4).

Figure 9.4: Sticking probability versus kinetic energy for the system  $H_2/(2 \times 2)S/Pd(100)$  for the analytical PES and a neural network PES based on the eleven configurations in Fig. 9.3. Neural Network: 9–20–20–1 *tl*. Parameter:  $\lambda(0)=0.98$ ,  $\lambda_0=0.99906$ ,  $a_{th}=0.3$ .

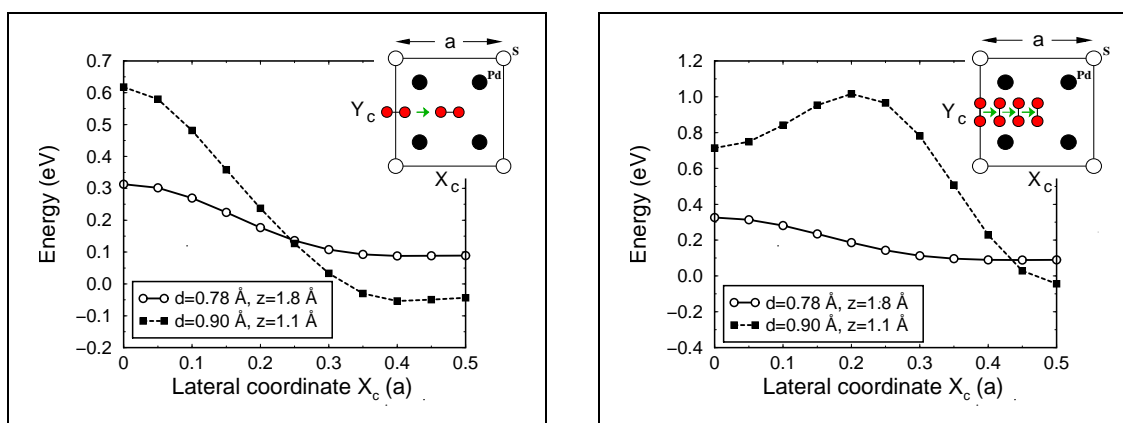


### 9.3.3 NN-fit based on eleven elbow plots and corrugation

In order to get a reliable description of dynamical properties for dissociation processes with neural networks it is not sufficient to follow the usual approach of restricting the calculations to 2D-cuts above high-symmetric sites. We need to add information about the PES which is not present in the elbow plots. Numerical calculations based on the analytical PES revealed that the steering effect is not only present on the clean Pd(100) surface, but also on the sulphur covered sample [51]. Molecules approaching the surface above sites with a high barrier to dissociation can be reoriented by the forces to more favourable adsorption configurations. The *distribution* of the barriers within the unit cell may be important for the reproduction of the steering effect and we have therefore tested how the incorporation of the energetic corrugation improves the interpolations.

The  $H_2$  molecule dissociates with its axis oriented parallel to the surface, the minimum path is located above the fourfold hollow site. Fig 9.5 displays the variation of the energy barriers the  $H_2$  molecule experiences during adsorption for two angular orientations above different lateral positions. In Fig 9.5(a) the molecule is oriented parallel to the surface above the sulphur bridge site with the hydrogen atoms pointing towards the fourfold hollow site. In order to scan the barriers we fixed the  $(Z_c, d, \theta, \phi)$  configuration for two different bond lengths  $d$  and heights  $Z_c$  and moved the molecule from the sulphur bridge site to the fourfold hollow site. The same is done in Fig 9.5(b) but now the H-atoms point initially in the direction of the sulphur atoms.

The configuration of the hydrogen molecule for the solid lines in Fig 9.5 correspond to the position of the maximum barrier in the entrance channel above the fourfold hollow site. In Fig 9.5(a) the energy barrier decreases monotonically from a value of 0.3 eV above the sulphur bridge site at  $(X_c, Y_c) = (0a, 0.5a)$  to 0.1 eV above the fourfold hollow site at  $(0.5a, 0.5a)$ , where “ $a$ ” defines the length of the  $(2 \times 2)$  unit cell.



(a) Hydrogen bond axis parallel to  $X_c$

(b) Hydrogen bond axis perpendicular to  $X_c$

Figure 9.5: Corrugation of the energy barriers for the system  $H_2/(2 \times 2)S/Pd(100)$  with  $H_2$  in two different orientations and its axis parallel to the surface. The energies are calculated for two different heights and bond lengths of the molecule. In both plots the molecule is moved from the sulphur bridge site to the fourfold hollow site.

The steady decrease of the energy barriers enables the molecule to approach the most favourable dissociation configuration above the hollow site even when it approaches the surface above, say, the palladium bridge site at  $(X_c, Y_c) = (0.25a, 0.5a)$ . If we further stretch the bond length of the hydrogen molecule and decrease the distance to the surface we obtain again a monotonic decrease of the energy (see the dashed line in Fig 9.5(a)). However, the energy barrier at the sulphur bridge site has significantly increased due to the shorter distance to the repulsive sulphur atoms. Above the fourfold hollow site the energy is now negative, the molecule has started to dissociate. If we let the bond length stretch further and allow the atoms to approach the surface the energy at the fourfold hollow site would further decrease reflecting that the dissociation process even on the sulphur covered Pd(100) surface is exothermic.

In Fig 9.5(b) the hydrogen molecule has been rotated by  $90^\circ$  in the azimuthal direction. For the configuration corresponding to the solid line again the energy barriers decreases monotonically as a function of the distance from the sulphur atoms. With a stretched bond length of  $d = 0.9 \text{ \AA}$  and a distance from the surface of  $z = 1.1 \text{ \AA}$  the picture has changed. Now the barriers are at its highest value above the Pd bridge site. This is due to the repulsive character of the palladium atoms at which the hydrogen atoms point at in this configuration. Again, at the fourfold hollow site the potential energy is already negative.

To improve the neural network approximation and to facilitate its ability of representing the steering effect we included the information about the variation of the energy barriers within the unit cell from Fig 9.5 in the training examples. We added 66 potential energies related to the corrugation of the barriers to the information governed from the previously discussed eleven 2D cuts. Namely, instead of optimising the neural network with 1189 and 1778 training examples based on the elbow plots only as shown previously in Fig. 9.4, p. 95, we use 1255 and 1844 points, respectively. The sticking probability for both training sets in Fig 9.6 agrees now semi-quantitatively with the underlying analytical potential-energy surface. Thus, incorporating only a small number of additional information to the calculated elbow plots can lead to significant improvement of the dynamical result.

Figure 9.6: *Sticking probability versus kinetic energy for  $H_2/(2 \times 2)S/Pd(100)$  calculated from the analytical PES and two neural network PESs. The neural PESs are based on the eleven configurations in Fig. 9.3 and the corrugation in Fig. 9.5*

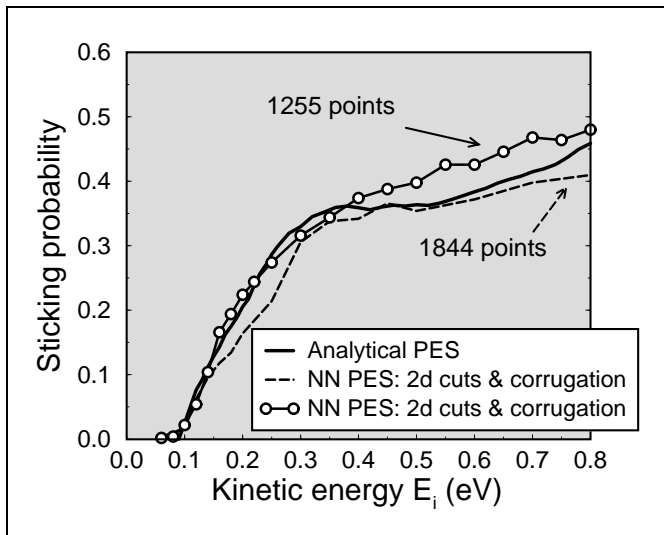


Fig 9.6 demonstrates that the incorporation of available physical knowledge about the system of investigation improves the interpolation considerably. Today it is well known that steering in dissociation dynamics is present and can be essential for the calculation of a dynamical property like the sticking probability. It is clear that the reorientation of the molecule is affected by the distribution of the energy barriers on the surface. Together with the knowledge about the favourable dissociation configuration of the studied molecule which can be gained from DFT-calculations we were able to calculate a small number of additional energies. With this new information the neural network was able to reproduce the adsorption coefficient with an error of less than 6%.

To conclude, we described that the fitting of a strictly activated PES with neural networks is significantly easier than the fit of a detailed PES with a coexistence of activated and non-activated pathways. In order to obtain reliable dynamical properties without any *a priori* knowledge of the reaction besides the symmetry a dense grid of points in all degrees of freedom is necessary. We pointed out that the number of energies required for extensive molecular dynamics with neural network is orders of magnitude smaller in comparison to on-the-fly *ab initio* dynamics. However, a number of  $10^4 - 10^5$  *ab initio* calculations still exceeds the currently available computer power. Additionally, we discussed that the usual approach of calculating  $10^2 - 10^3$  energies within two-dimensional cuts above high-symmetric sites is not sufficient for a neural network application. On the other hand, we demonstrated that the incorporation of physical knowledge corresponding to a small amount of additional information leads to a reliable description of the process.

## 9.4 Constrained Neural Network fitting approach

So far we have fitted the complete six-dimensional potential-energy surface in one step. Yet, it might also be possible to split the fitting into separate parts or to use a combination of different methods to approximate a PES.

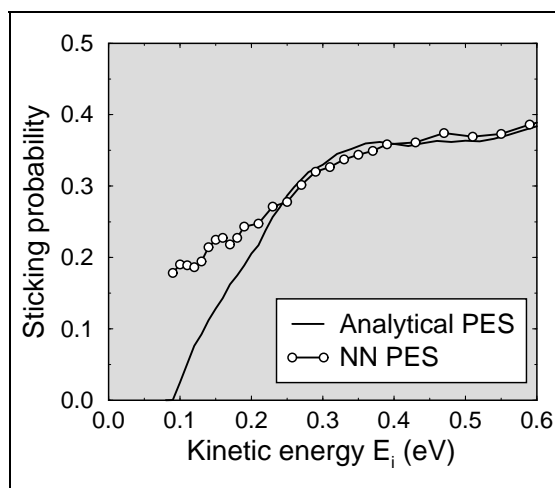
For instance, a hybrid scheme using a combination of numerical and analytical interpolation techniques has been proposed recently for the hydrogen dissociation processes [48, 28, 27]. The method constructs the six-dimensional PES for diatomic molecule-surface interactions from the low dimensional cuts obtained in *ab initio* calculations. The authors developed a *corrugation-reducing* procedure based on the consideration that most of the corrugation in a molecule-surface PES is already embedded in the atom-surface interactions. The subtraction of the latter leads to a much smoother function, which is easier to interpolate. The resulting PES is then approximated by a combination of numerical and analytical methods. The two dimensional *Zd* cuts are determined and interpolated by splines, whereas the corrugation in the lateral directions is described by a Fourier expansion. The proposed method has been applied to the dissociation of hydrogen on Pd(111) [48, 28], Ni(111), (100) and (110) [27].

We followed a different approach. It is based on the physical consideration that a minimum path towards dissociation exists. On the clean Pd(100) this is the Pd bridge site with the H-atoms pointing to the hollow sites, on the sulphur covered surface it is the fourfold hollow site with the atoms directed towards the bridge sites. All other

2D cuts have a *higher* energy barrier in the region of the cut where the bond breaks. The idea behind our approach is the recognition that the minimum energy barrier on an activated surface is very important as it determines the onset of the sticking probability. The molecule needs the corresponding kinetic energy to overcome this barrier before it can dissociate. A neural network is not explicitly “aware” of the existence of a minimum barrier. In principle, the numerical approximation is free to put a smaller barrier somewhere else in a two-dimensional cut which has not been determined by *ab initio* methods. This will of course have important dynamical consequences as can be seen from the neural network sticking curve in Fig. 9.7. At low kinetic energies the adsorption probability is much too high because the interpolation produced pathways - which have not been calculated by density functional theory methods - with a lower energy barrier than the determined minimum value. This problem vanishes automatically if a very high number of *ab initio* data is available, as discussed previously. Yet, with the low number of information available at the moment one always has to be very careful in judging the accuracy of the fit. Hence, in order to increase the reliability of the interpolation it will be advantageous to incorporate the available physical knowledge about the energy barriers and to fix the minimum barrier in a neural network interpolation.

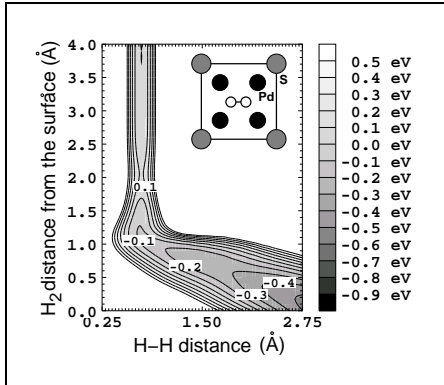
In addition, we observed that the *ab initio* calculated elbow plots either on the clean or the sulphur covered Pd(100) surface look rather similar. They differ mainly in the region where the bond breaks (see for instance Fig. 8.1, p. 74, H<sub>2</sub>/Pd100 and Fig. 9.1, p. 92). Therefore it might be useful to present the network first the general shape of these two dimensional cuts before fitting the details of the complete six dimensional problem. A neural network having *learned* to represent one of the elbow plots may need only little additional information, e.g. the energy in the bond-breaking region, to fit the PES above the other adsorption sites.

The motivation of the constrained fitting approach is therefore twofold: How can we perform a neural network interpolation which (a) learns first the general shape of the elbow plots before it tackles the complete 6D problem, and (b) at the same time reflects a lower bound of the energy barrier, the minimum barrier, for all possible adsorption configurations? In order to do so we split the fitting of the PES into two parts using two neural networks. In a first step we fit the most valuable elbow plot, the minimum pathway to dissociation with the minimum energy barrier, i.e. we perform a 2D neural network interpolation. In the second step we fit the complete 6D problem under the requirement that all other elbow plots have a higher energy than the previously interpolated one. In detail, the constrained fitting approach looks as follows:



**Figure 9.7:** Example of a neural network sticking curve where the minimum barrier has not been fitted correctly.

### Step I - 2D Fit



**Figure 9.8:** The minimum pathway for the dissociation of hydrogen on the sulphur covered Pd(100) surface.

To represent the general shape of the elbow plots we first perform a 2D interpolation of the most favourable adsorption configuration with the minimum barrier for the reaction. Hence, we use a neural network with two input nodes to interpolate the minimum pathway towards dissociation:

$$V_{2D}^{NN}(\{a_w\}, Z, d) \equiv \text{minimum pathway},$$

where  $Z$  is the distance of the molecule from the surface,  $d$  is the bond length, and  $\{a_w\}$  is the set of neural network weights, which have to be optimised. For the dissociation of  $\text{H}_2$  on a  $(2 \times 2)\text{S}/\text{Pd}(100)$  surface the minimum barrier above the hollow configuration in Fig 9.8 is 0.1 eV.

### Step II - 6D Fit

Having approximated the minimum dissociation pathway and determined the set of weights  $\{a_w\}$  we now want to tackle the full problem under the assumption that all other 2D cuts can have a higher energy only. We write the following equation for the six-dimensional problem:

$$V_{PES}^{NN}(\{b_w\}, X, Y, Z, d, \theta, \phi) = V_{2D}^{NN}(\{a_w\}, Z, d) + [V_{6D}^{NN}(\{b_w\}, X, Y, Z, d, \theta, \phi)]^2. \quad (9.1)$$

This is the constraint under which we perform the 6D approximation of the PES. We optimise the corresponding set of weights  $\{b_w\}$  only, whereas the weight vector for the elbow plot  $\{a_w\}$  remains fixed. The square of the six dimensional neural network potential  $V_{6D}^{NN}$  ensures that the energy in all other 2D cuts depending on  $Z$  and  $d$  can only be higher than the energy within the minimum path  $V_{2D}^{NN}$ . However, we point out that this equation is not fully motivated by physical considerations. It is true that a minimum barrier for the dissociation process exists, but in Eq. (9.1) we required slightly more. The constraint is based on the assumption that the energy in all other configurations than the most favourable one is higher for *all values* of the bond length  $d$  and the distance from the surface  $Z$  in the  $Zd$  plane and not only for *one pair* of distances  $(Z, d)$  at the location of the barrier. We discuss this assumption in more detail at the end of this section.

Following the proposed procedure we interpolated the fourfold hollow site configuration using 125 data points and a  $2-20-20-1$  *tl* neural network as shown in Fig. 9.9(a). Based on this interpolation the complete PES has been approximated under the constraint of Eq. (9.1) with a  $9-20-20-1$  *tl* net. For this purpose we choose a small number of 165 energies in the other ten adsorption configurations, see Fig 9.9(b). Additionally, we included 88 total energies describing the corrugation of the barriers within the unit cell as discussed previously. The new neural network has been applied



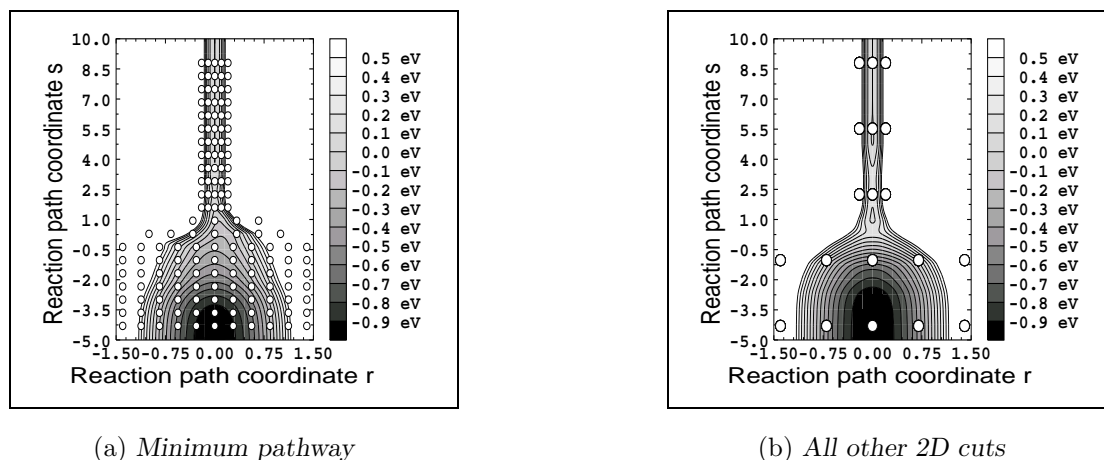
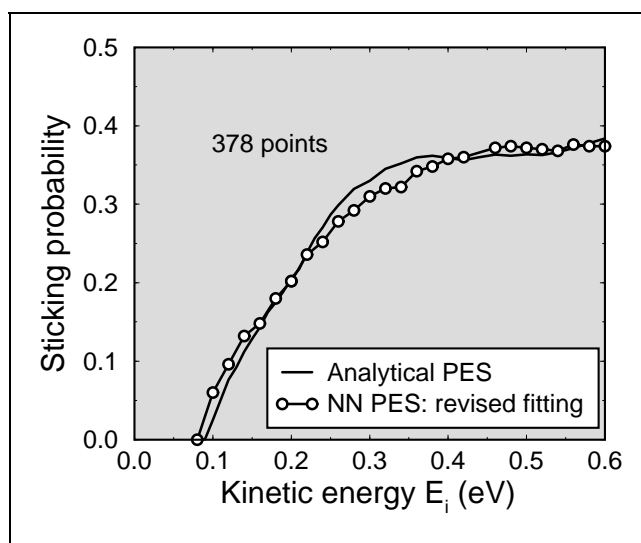


Figure 9.9: Data sampling (circles) for the constrained fitting approach in reaction path coordinates along ( $s$ ) and perpendicular ( $r$ ) to the reaction path. (a) The NN minimum path is based on a dense grid. (b) The sampling in the other 2D cuts is sparse.

to classical molecular dynamics calculations and the sticking probability as displayed in Fig. 9.10 has been determined. The agreement with the underlying analytical PES is very good, the differences being smaller than 3%. To check our procedure we have repeated the constrained fitting approach with a slightly different training set of 348 points and achieved similar accuracy.

This demonstrates that the coordinates for a dissociation process of a diatomic molecule can be efficiently separated. A neural network having learned the shape of the elbow plots previously can very accurately reproduce other configurations with only small additional information. We are able to describe the full dissociation problem reliably with only a small number of *ab initio* calculations ( $< 500$ ).

Figure 9.10: Constrained neural network fitting approach. In the first step the minimum path has been approximated by a 2–20–20–1 *tl* network based on a training set of 125 energies. Secondly, the 6D PES under the constraint of Eq. (9.1) has been determined with a 9–20–20–1 *tl* net and only 378 examples.

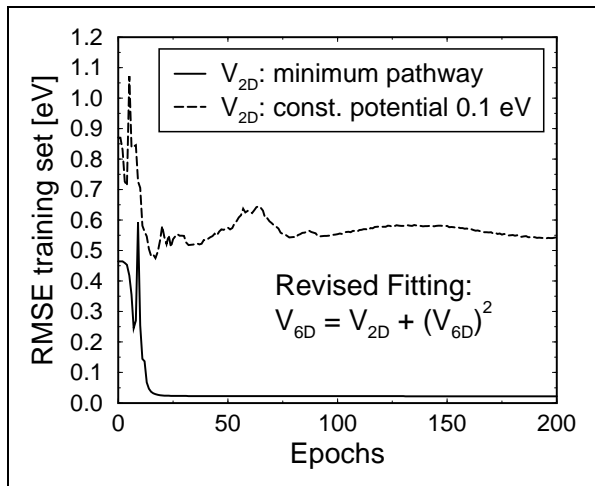


### Discussion of the constraint Eq. (9.1)

The constraint formulated in Eq. (9.1) proved to be very powerful. However, as already mentioned, the equation is not fully based on physical considerations. We assumed that the energy in all other configurations than the minimum pathway is strictly higher for *all*  $(Z, d)$  values and not only for *one pair of distances*  $(Z, d)$  at the location of the barrier. We shall further elaborate this issue. What if the assumption is wrong and the fitted 2D plot in the first step is not the lower energy bound for all other 2D geometries? Does the constrained fitting approach reflect in a way that it is based on a wrong assumption? In order to test this let us consider the case that we have not fitted the minimum pathway, but for instance a constant potential with the energy of the minimum barrier, i.e.:

$$V_{2D}^{NN}(\{a_w\}, Z, d) = 0.1 \text{ eV} \quad \forall Z, d. \quad (9.2)$$

If we use this constant potential in the first step of the constrained fitting approach and try now to approximate the full problem under the constraint of Eq. (9.1), it will be impossible to achieve a new set of weights  $\{b_w\}$  which minimises the error to the desired accuracy. Since Eq. (9.1) assumes that the energy for all  $(Z, d)$  values can only be higher, no accurate interpolation reflected by the root mean squared error will be possible. The energy in the different elbow plots is not strictly higher than



**Figure 9.11:** Test of the constrained fitting approach using different two-dimensional potentials in Eq. (9.1). Plotted is the RMSE training error versus the number of epochs. A wrong 2D potential leads to a high level of training error for the complete 6D PES.

a wrong constraint. Normally, it is *always* possible to minimise the *training* error close to zero, the level of error is just a function of the number of parameters. The problem is rather to minimise the *test* error to a desired value. Here, with the use of the constraint in Eq. (9.1), a wrong assumption is reflected by a high error on the *training* examples and can therefore be spotted easily. For comparison, in the previously

0.1 eV. For instance, the potential is of course lower in the gas phase far away from the surface and also close to the surface the molecule gains energy on dissociation, the potential is even negative. The assumption formulated in Eq. (9.1) based on the constant two-dimensional potential in Eq. (9.2) is obviously wrong.

We have tested how the optimisation procedure of the neural network parameter vector  $\{b_w\}$  of the six-dimensional PES is affected if such a false potential is used. The training error for a 9–20–20–1 *tl* neural network using Eq. (9.1) and the constant 2D potential of Eq. (9.2) is shown in Fig. 9.11. The training error cannot reach any satisfactory level of error, because the minimisation is based on

discussed constrained fitting of hydrogen dissociation on the sulphur covered Pd(100) surface based on 378 potential energies we achieved a training error of 0.023 eV after a low number of 30 epochs also plotted in Fig. 9.11. This is well below the desired *ab initio* accuracy of 0.1 eV and reflects that the two-dimensional minimum pathway towards dissociation described by the analytical PES is indeed lower in energy than all other configurations and justifies the proposed assumption for this case.

So far we have discussed what happens if the assumption is wrong that the fitted 2D potential in the first step of the constrained fitting is a lower bound for all other 2D cuts. We have seen that this case is reflected by a high training error. But a necessary requirement of the constrained fit is that information about such a lower energy bound is indeed *available*. Since this work is concerned with interpolation and not extrapolation the main features of the PES, i.e. the minimum and maximum potential energies, should be present in the data set. In particular the minimum and maximum values within the 2D elbow plots should be available. Yet, these extrema do not have to be located within *one single* elbow plot. For instance, let us consider the problem that *none* of the 2D cuts is strictly lower in energy for *all*  $(Z, d)$  values than the rest of the cuts. For instance, there may exist an adsorption geometry where the fractions of the dissociation process, here the H-atoms, are bound more strongly to the surface than above the site with the minimum barrier to dissociation. In such a case the energy close to the surface would be lower in the former configuration than in the latter, whereas in the barrier region the situation might be reversed. How can we incorporate such a situation in the constrained fit? This is of no problem, since the constraint in Eq. (9.1) does not require that the lower 2D bound is associated with *one single* elbow plot. In fact, one can always construct such a lower bound from the detailed information of *all* elbow plots. One just performs a 2D interpolation of each elbow plot, which needs to be done in any case for the plotting of the elbow cuts, and gains information on a grid of  $Z$  and  $d$  values. From this one generates the 2D lower bound by taking the minimum energy at each  $(Z, d)$  pair from *all elbow plots*. This artificial minimum path then reflects the minimum energy of all calculated adsorption configurations and serves as an input for the first step of the constrained fitting approach  $V_{2D}^{NN}$  in Eq. (9.1).

The discussion of this paragraph can be summarised as follows:

- During the optimisation procedure of the constrained fitting approach a false two-dimensional lower energy bound  $V_{2D}^{NN}$  is reflected by a high training error.
- If none of the calculated elbow plots is a strictly lower bound to all other 2D cuts, one can generate this lower bound as follows:
  - (1) Perform a 2D interpolation of each calculated elbow plots.
  - (2) Take the minimum energy on a grid of  $Z, d$  values of **all** interpolated elbow plots. The generated set of values forms the lower bound.

A schematic representation of the constrained fitting approach of potential-energy surfaces with neural networks can be found in Fig. 9.12. We emphasise, that the presented application of two neural networks for the interpolation of a PES is tailored to the specific needs of dissociative reaction dynamics and the development of this approach required insight into the problem, which is not always available.

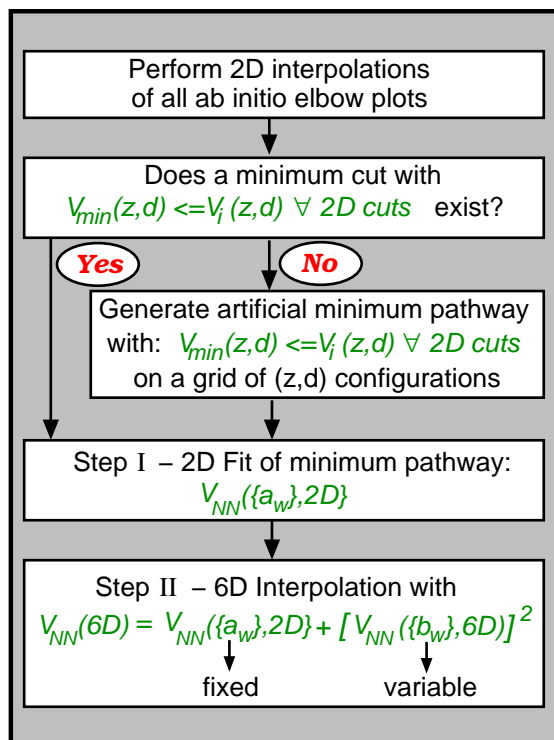


Figure 9.12: *Constrained Neural Network fitting approach for potential-energy surfaces of dissociation reactions. The fitting is based on two separate interpolations with neural networks. First, a real or an artificial minimum pathway is approximated before the complete six-dimensional potential energy is fitted.*

## 9.5 Energetic corrugation and its influence on adsorption probabilities

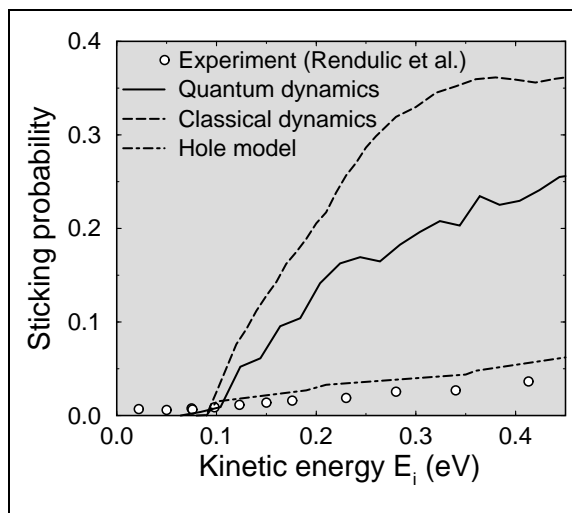
In the previous sections we discussed how the fitting and the sampling of the data have to be adapted in order to gain a reliable description of dynamical properties with neural networks even with a low number of data points. For instance, we have seen that for an accurate reproduction of the analytical sticking curve it is important to include information about the corrugation of the potential energy in the fit. Yet, in the introduction in Sec. 2.3.2 we pointed out that a discrepancy between this analytical sticking curve and the experimental one [4] exists. We will now discuss this difference in more detail. This section thus marks an end to the sole reproduction of an analytical PES of the previous two chapters. In the following we will use the neural network as a tool to study the influence of the corrugation on adsorption probabilities *in general*.

Fig. 9.13 compares the quantum mechanical and classical results of the sticking probability obtained from the analytical PES [33, 51] with the experiment [4]. The calculated sticking probabilities are significantly larger than the experimental results. It has been argued that these discrepancies might be caused by the presence of sub-surface sulphur in the experimental samples [51, 62]. In the experiment by Rendulic et al. the sulphur adlayer was obtained by heating the sample, which leads to the segregation of bulk sulphur to the surface. Since the determination of the sulphur coverage at and below the surface is not very accurate, it might well be that a certain fraction of sub-surface sulphur is still present in the experimental samples assuming that the segregation was not fully complete. This will have important consequences

for the surface reactivity because the species below the surface will bind to the Pd atoms and hence make the binding of H-atoms to the surface Pd atoms more difficult. Both the experiment [4] and the DFT calculations [29] show that the  $H_2$  dissociation on the sulphur-covered Pd(100) surface depends sensitively on the actual sulphur coverage. In an independent experiment Burke and Madix [140] showed for a higher coverage of  $\theta_S = 0.5$  that  $H_2$  dissociation at this surface is completely suppressed at thermal energies, whereas Rendulic et al. still observed dissociation even at this coverage [4]. Therefore some uncertainties concerning the preparation of the sulphur adlayer on Pd(100) in the experiment remains.

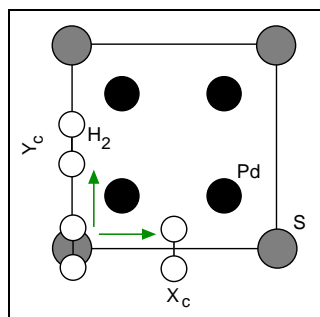
Let us now focus on the differences between the two calculated sticking curves. The classical treatment of hydrogen dynamics overestimates the sticking probability compared to the quantum mechanical results. For energies larger than the minimum energy barrier the classical adsorption probability rises to values that are almost 50% larger than the respective quantum probabilities. The suppression of the quantum result is caused by the strong corrugation and the anisotropy of the PES. The minimum barrier is rather localised in the lateral coordinates parallel to the surface and also in the rotational degrees of freedom of the molecule. Hence, the associated modes become “frustrated” and the interaction of a  $H_2$  molecule with the substrate gives rise to zero-point energies. These energies need to be taken from the incident beam and therefore the effective kinetic energy of the impinging molecule is reduced. Consequently the fraction of the molecules which are able to overcome a certain barrier towards dissociation is reduced and the sticking probability is lower. However, the overall trend of the sticking curve with increasing kinetic energy is well reproduced also by classical dynamics.

In addition, also the integrated barrier distribution or the so-called “hole model” is plotted in Fig. 9.13. It is the fraction of the configuration space for which the dissociation barrier is less than the initial kinetic energy of the molecule. Since, e.g., pathways associated with an upright molecule are purely repulsive, the fraction of the configuration space in which the molecule would be able to dissociate without changing its configuration is rather small. The dynamical results in Fig. 9.13 are much larger than the integrated barrier distribution. Groß et al. found that also on the S covered Pd(100) surface steering to configurations and sites with lower barriers [51,62] is effective, which results in the higher sticking at a given energy. Hence, the adsorption probability probes the very details of the corrugation of the PES and not only the barriers corresponding to the initial configuration of the impinging molecule.



**Figure 9.13:** Sticking probability versus kinetic energy for a  $H_2$  beam on a  $(2 \times 2)S/Pd(100)$  surface. Theory: quantum and classical dynamics on an analytical PES [51]. Experiment: from [4].

It is striking that the integrated barrier distribution, where no steering is present, agrees much better with the measured probabilities than the dynamical result. Could it be, that the re-orientation of the molecules is overemphasised by the analytical interpolation for this system? Since the corrugation is connected with the steering effect and thus with the difference between the hole model and the dynamical result and also between the classical and quantum adsorption probabilities, we will now study its influence in more detail. As an example we will take the corrugation of the  $\text{H}_2$  molecule in its most favourable dissociation configuration with the molecular axis parallel to the surface. Due to the steering effect also molecules approaching the surface in an upright position will be very efficiently re-oriented and re-directed by the forces towards the parallel configuration. The corresponding distribution of energy barriers for this geometry will therefore influence the probability for sticking of the molecules.



**Figure 9.14:**  $\text{H}_2$  geometries for Fig. 9.15

In order to visualise the analytical variation of the energy in the lateral directions of the surface we use the same approach as presented in the discussion of the clean  $\text{Pd}(100)$  surface on page 87. We fix the  $\text{H}_2$  molecule in its most favourable dissociation configuration, i.e. parallel to the surface, at a position which corresponds to the region where the H-H bond starts to break, and move the molecule in this position across the unit cell. Namely, we scan the barrier heights by fixing the distance of the molecule from the surface at  $Z = 1.9 \text{ \AA}$  with a bond length of  $d = 0.76 \text{ \AA}$  at angles  $(\theta, \phi) = (90^\circ, 0^\circ)$  and vary the lateral coordinates  $X_c$  and  $Y_c$  as illustrated in Fig. 9.14.

The energy as a function of the lateral coordinates calculated from the analytical PES is shown in Fig. 9.15(a). The location of the surface Pd and S atoms have been indicated by the dark filled circles. If we look e.g. at the variation of the energy along the diagonal from the sulphur on-top to the hollow site in the middle of this contour plot, we see that the analytic energy first drops rather steeply reaching values of 0.5 eV shortly behind the Pd atom and then levels out smoothly to a value of 0.1 eV, i.e. the minimum barrier above the hollow site. 20% of the area of the unit cell has an energy barrier lower than 0.2 eV rising to 33% for an energy  $\leq 0.3$  eV. Hence, if the kinetic energy exceeds the value of the minimum energy barrier a rather large area of the surface becomes soon available for sticking. For instance, molecules approaching the surface upright over a Pd atom with a low kinetic energy do not need to be re-directed on the surface very far after being reoriented to the favourable configuration in order to dissociate. This form of the corrugation is therefore connected to the sharp increase of the sticking probability based on the analytical PES in Fig. 9.13. In addition, we have also included the calculated *ab initio* energies in Fig. 9.15(a) marked by the white filled squares above the high symmetric configurations, the fourfold hollow site, the Pd bridge site, the other Pd hollow site between the sulphur atoms, and the sulphur on-top position. This sparse set of energies has been interpolated analytically [33] by using Fourier terms in the lateral directions to the form presented in Fig. 9.15(a).

Inspired by the fact that the integrated barrier distribution, where steering is not present, gives good agreement with the molecular beam experiment, we will now discuss

a modified interpolation of the sparse set of *ab initio* energies where the steering effect might be suppressed. Let us consider the case that the potential energy increases more steadily from the fourfold hollow site to the sulphur on-top site, while at the same time the energies of the *ab initio* points are kept fixed as plotted in Fig. 9.15(b)? The area of the unit cell corresponding to a barrier lower than 0.3 eV is now reduced to half of the value of the original analytical interpolation. On such a potential-energy surface the valley in which the molecules have to be steered to for dissociation is much narrower. The forces acting on a hydrogen molecule approaching the sulphur or palladium on-top sites are now considerably smaller than the corresponding values on the original analytical PES. Thus, the re-orientation and re-direction of the impinging molecules should be much less pronounced and the steering effect might be less efficient. In turn, this could lead to a reduced dissociation probability.

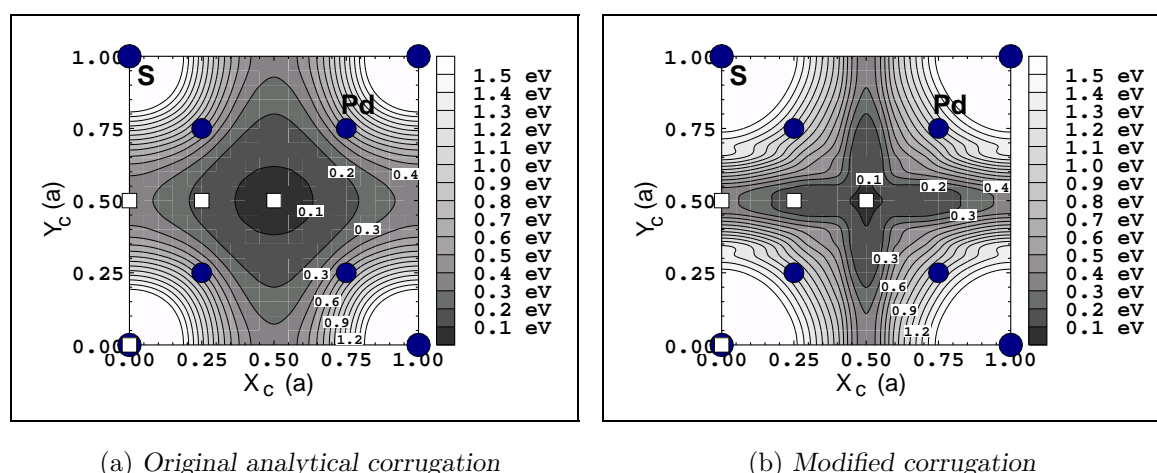


Figure 9.15: *Corrugation of the energy across the unit cell. The molecule is oriented parallel to the surface with its bond axis parallel to  $Y_c$  at a height  $z=1.9 \text{ \AA}$  with a bond length  $d=0.76 \text{ \AA}$  as plotted in Fig 9.14. (a) and (b): potential energies as a function of the lateral coordinates ( $X_c$ ,  $Y_c$ ). The contour spacing is 0.1 eV. The filled circles mark the location of the surface atoms, the white squares the calculated *ab initio* points.*

For a test of the influence of the corrugation we performed two separate neural network interpolations based on two different data sets. First, we fitted 496 energies obtained from the analytical PES including 72 points corresponding to the original corrugation of Fig. 9.15(a). Secondly, we used the same 496 configurations but changed the values of the 72 points associated with the corrugation in accordance to the modified corrugation of Fig. 9.15(b). For both interpolations we applied the constrained fitting approach using the same 8–10–10–10–1 tl neural network. In order to check the accuracy of both fits we applied the identical test set with 117 energies above the usually calculated high symmetric sites. The neural network interpolations based on the two different training sets achieved both a training error of 50 meV and a test error of 0.20 eV after 500 epochs. The higher errors occur only for energies where the potential is already elevated, i.e. not in the region of the valley of each elbow plot. The latter region has been weighted 10 times higher during the interpolation.

Fig. 9.16 shows the adsorption probabilities obtained from the two neural network fits compared to the analytical and the experimental probabilities. The neural network sticking curve based on the original analytical PES reproduces indeed the analytical sticking curve very well. Yet, the sticking probability corresponding to the neural network PES trained with the modified corrugation is significantly lower than the analytical results and closer to the experimental ones. At an energy of 0.3 eV the corresponding adsorption probability is less than half the value of the original analytical PES, but also still three times as large as the measured result. However, we should keep in mind that quantum dynamical calculations on such a corrugated PES will further reduce the results considerably and therefore would further decrease the discrepancy with the experiment.

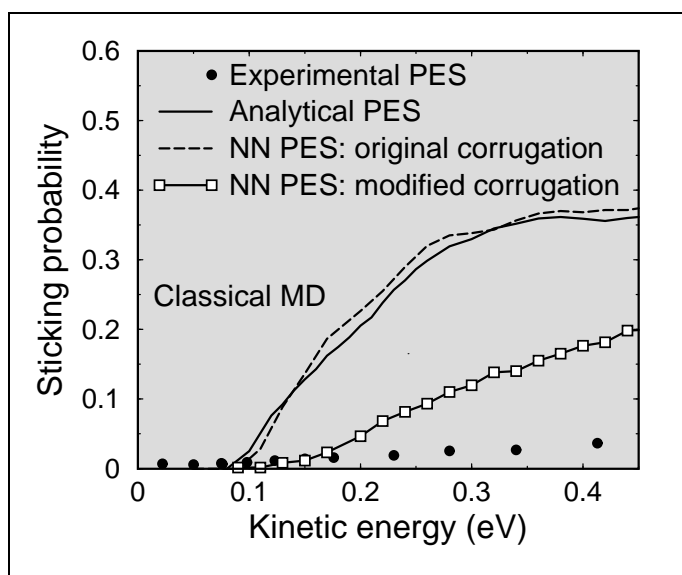


Figure 9.16: Adsorption probabilities as a function of the kinetic energy of hydrogen dissociation on a sulphur covered  $\text{Pd}(100)$  surface. Comparison of two neural network sticking curves based on the original corrugation of Fig. 9.15(a) and the modified corrugation of Fig. 9.15(b) with the analytical [51] and the experimental result [4].

Hence, the form of the corrugation influences the dynamical result greatly and in addition to the presence of sub-surface sulphur could also be a source of the discrepancy between experiment and theory. It has been found theoretically that the poisoning effect of sulphur on the  $\text{Pd}(100)$  surface is due to a mixture of a direct interaction between the adatom and the hydrogen molecule and a sulphur-induced modification of the local electronic structure at the surface  $\text{Pd}$  atom [33]. A functional form of such interaction is not obvious. Without a calculation based on DFT methods it will be difficult to judge which corrugation is more likely. We therefore recommend to always extend the theoretical mapping of potential-energy surfaces from the calculation of elbow plots in the  $Zd$  plane above high-symmetric sites to two-dimensional cuts in the lateral directions of the surface.

We will apply this procedure in the following chapter concerning the *ab initio* calculation of the PES for the dissociation of hydrogen on a potassium covered  $\text{Pd}(100)$  surface. We will then come back to the discussion of this section.



Part III.  
Applications: Neural Network fits  
to *ab initio* data



---

## Chapter 10

### Neural Network application: Hydrogen dissociation on $\text{K}(2 \times 2)/\text{Pd}(100)$

---

Having tested the neural network approach of fitting potential-energy surfaces based on analytical functions we will now present its application to *ab initio* data: The dissociation of hydrogen on a potassium covered Pd(100) surface. Experiments by Solymosi and Kovács have shown that not only sulphur adatoms, but also potassium adsorbates poison the  $\text{H}_2$  dissociation reaction [34]. This is of special interest since commonly electronegative atoms of groups Vb and VIb poison a catalytic process, whereas the electropositive alkali-metal atoms promote the process, or vice versa. A density functional theory study by Wilke and Cohen revealed that potassium hinders  $\text{H}_2$  dissociation by forming energy barriers in the entrance channels, i.e. at large distances from the surface, where the intramolecular bond is not significantly stretched [37]. In the exit channels, when the molecules are dissociated, the presence of potassium atoms tends to stabilise the bond of the hydrogen atoms with the surface. The theoretical investigation focused on two dissociation pathways with the hydrogen molecule above hollow sites.

We have now extended this previous work and investigated in detail how the poisoning of the  $\text{H}_2$  dissociation on Pd(100) due to the presence of potassium depends on position and orientation of the molecule. Overall we calculated eleven elbow plots with the molecule above high-symmetric sites. Moreover, we analysed the corrugation of the PES in even more detail by collecting information of the potential energy as a function of the lateral coordinates within the surface unit cell. The *ab initio* energies have been fitted by neural networks in two different ways. First, we performed an interpolation of the six-dimensional PES in one step with a single neural network. Secondly, we applied the previously introduced constrained fitting approach and employed two different neural networks in the approximation. The neural network PESs have been utilised in classical molecular dynamics calculations and the sticking probability of the dissociating  $\text{H}_2$  molecule as a function of the kinetic energy has been determined.

The structure of the chapter is as follows. We will first introduce experimental findings for hydrogen dissociation on potassium covered palladium surfaces. We will then describe the computational details of the density functional theory calculations before we discuss the *ab initio* PES. Finally, we present the neural network interpolations of the potential energy and their application to molecular dynamics.

## 10.1 Hydrogen on K/Pd surfaces in experiments

It has been found experimentally that potassium adatoms form two ordered structures on the Pd(100) surface, a p(2×2) corresponding to a potassium coverage of  $\theta_k = 0.25$  and a c(2×2) at  $\theta_k = 0.5$  [141, 142]. The potassium equilibrium adsorption positions are surface hollow sites which hydrogen also prefers to occupy. Potassium adatoms induce a strong decrease of the metal work function. For the p(2×2) the work function is about 4 eV lower than for the clean surface.

The dissociation of H<sub>2</sub> and D<sub>2</sub> on the K-covered Pd(100) surface has been studied by Solymosi and Kovács using thermal desorption spectroscopy (TDS), work-function changes and ultraviolet photoelectron and Auger spectroscopy [34]. At a potassium coverage of 25% and a temperature of 100 K the sticking probability is 30 times less than that measured for the clean surface. At this temperature the energy distribution is dominated by low-energy molecules which are particularly sensitive to energy barriers for dissociation. The experiments indicated that on K-covered Pd(100) surfaces, dissociative adsorption of H<sub>2</sub> molecules is kinetically hindered. Whereas the sticking probability is reduced, at the same time TDS experiments showed a monotonic increase of the temperature corresponding to the maximum desorption rate of hydrogen with potassium coverages [34]. This was interpreted as an indication for a stabilisation of H-atoms by the presence of potassium co-adsorbates. Such a simultaneous poisoning of H<sub>2</sub> dissociation and stabilisation of the H-metal bond on the K-covered Pd(100) surface is at variance from the case of S-covered Pd(100), which we discussed previously. For the latter system both the dissociation probability of hydrogen molecules and the strength of the H-metal bond are reduced. Beam experiments revealed that also on the Pd(110) surface pre-adsorbed potassium acts as an inhibitor to adsorption [143]. However, to our knowledge sticking coefficients for a beam of hydrogen molecules on a K/Pd(100) surface have not been measured up to now.

## 10.2 *Ab initio* potential-energy surface

### 10.2.1 Computational details

We have used density-functional theory together with the generalised gradient approximation (GGA) [91] for the exchange-correlation functional. The relativistic Kohn-Sham equations are solved with the full-potential linear augmented plane-wave (FP-LAPW) method [82, 83]. The parameters used in the FP-LAPW code were similar to those used in a previous study of the same system by Wilke and Cohen [37]. The FP-LAPW wave functions in the interstitial region are represented using a plane wave expansion up to  $E_{max}^{wf} = 13$  Ry. Tests of the convergence of the results have been performed with a higher cutoff of up to  $E_{max}^{wf} = 15$  Ry, but led to energy difference of less than 20 meV. Due to the small muffin-tin radius around the H atoms of  $R_{MT}(H) = 0.32$  Å in accordance with the H<sub>2</sub> equilibrium bond length of 0.75 Å, it is necessary to take into account plane waves up to  $E_{max}^{pot} = 169$  Ry for the potential representation. Inside the muffin-tin spheres the wave functions are expanded in spherical harmonics with  $l_{max} = 10$ , and non-spherical components of the density and potential are included up to  $l_{max} = 3$ . The

$k$ -integration used 32 uniformly spaced points in the two-dimensional Brillouin zone corresponding to the  $(2 \times 2)$  surface unit cell. Wilke and Scheffler [22] tested the interaction of the dissociating hydrogen molecules of neighbouring unit cells on the clean Pd(100) surface using different surface structures. They found that a  $p(2 \times 2)$  unit cell is sufficient to ensure negligible residual-interaction.

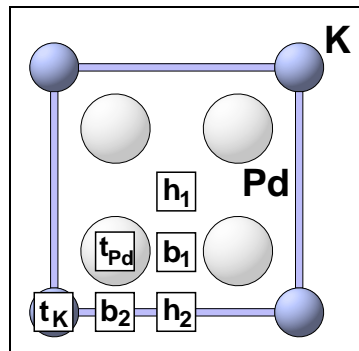
The metal substrate was modelled by five layers separated by a 13 Å thick vacuum region. Adatoms are symmetrically adsorbed on both sides of the slab. Wilke and Cohen studied the influence of the slab thickness on the potential energy for slabs with up to seven layers. They found energy differences change by less than 50 meV [37]. We have tested the influence of the size of the vacuum region by expanding it up to a value of 20 Å. The energy differences changed again only by less than 50 meV.

The calculated lattice constant of bulk Pd is 3.95 Å which is 1.5% larger than the experimental value and 2% smaller than the value obtained by a non-relativistic calculation of the same system [37]. The smaller lattice constant and the better agreement with the experiment of the relativistic calculation is due to the nature of the Pd d-states, which are highly localised and therefore should be treated relativistically. The K-adatoms were placed into the surface hollow sites of the Pd(100) surface [141]. The calculated adsorption height of potassium forming a  $p(2 \times 2)$  structure is  $h_0^K = 2.41$  Å which is in agreement with previous results [37]. The energy zero is taken as the energy of the geometry where the molecule is sufficiently far away from the surface with  $Z_c = 5.05$  Å, i.e. where the molecule and the surface do practically not interact. The calculated equilibrium bond length in this configuration is 0.75 Å, which is in agreement with the experimental value.

### 10.2.2 Potential-energy surface

Neglecting the surface relaxation effects, the potential-energy surface for the dissociative adsorption of a hydrogen molecule over a potassium covered Pd(100) surface is six-dimensional. As discussed previously, the six dimensions correspond to the six degrees of freedom of the  $H_2$  molecule  $X_c, Y_c, Z_c, d, \theta, \phi$ . In order to map this high-dimensional PES we followed the usual approach of computing 2D cuts through the configuration space as discussed in Sec. 3.3, p. 33. The cuts are defined by the site  $(X_c, Y_c)$  and the orientation  $(\theta, \phi)$ . Figure 10.1 shows the  $(2 \times 2)$  surface unit cell. Different adsorption sites within the unit cell exist, namely two bridge sites  $b_1, b_2$  two hollow sites  $h_1$  and  $h_2$ , the top sites above palladium atoms  $t_{Pd}$ , and above potassium atoms  $t_K$ .

Overall, we have calculated 659 *ab initio* energies. The majority were sampled within eleven elbow plots above the high-symmetric sites. These are the same 2D cuts which we used in the discussion of the sulphur-covered Pd(100) surface. Additionally, we focused particularly on the energies associated with the corrugation of the energy barriers across the unit cell.



**Figure 10.1:**  $(2 \times 2)$  K-covered Pd(100) surface with two hollow sites  $h_1, h_2$ , bridge sites  $b_1, b_2$  and top sites  $t_{Pd}, t_K$ .

To distinguish between the different orientations we will characterise them by the position of the H atoms and the centre of mass of the molecule. For instance, a geometry  $\mathbf{h}_2\text{-}\mathbf{b}_1\text{-}\mathbf{h}_1$  denotes the situation where the centre of mass is over the bridge position  $\mathbf{b}_1$  and the H atoms are oriented towards the hollow sites  $\mathbf{h}_1$  and  $\mathbf{h}_2$ , respectively.

### I. Two-dimensional cuts through the PES at hollow sites

Wilke and Cohen [37] focused in their study of the same system on two adsorption geometries  $\mathbf{b}_1\text{-}\mathbf{h}_1\text{-}\mathbf{b}_1$  and  $\mathbf{b}_1\text{-}\mathbf{h}_2\text{-}\mathbf{b}_1$  above hollow sites. They found that a potassium  $\text{p}(2 \times 2)$  adlayer induces energy barriers for the dissociation of hydrogen in the entrance channel of the PES, hindering the approach of the molecules to the surface. The potential-energy surface changes considerably in comparison to the clean surface. Whereas on the latter surface the reaction was non-activated, on a K-covered surface all pathways to dissociation are now activated. Their calculations showed that the energy along the reaction pathways above the different hollow sites differ only marginally. The barrier height is 0.18 eV, whether the  $\text{H}_2$  molecule dissociates over the central hollow site  $\mathbf{h}_1$  or over the hollow site  $\mathbf{h}_2$ . The top of the barrier was found in the entrance channel at  $Z_c = 1.85 \text{ \AA}$ , with the H-H bond not yet stretched.

Figure 10.2(a) shows the recalculated PES for the  $\mathbf{b}_1\text{-}\mathbf{h}_1\text{-}\mathbf{b}_1$  dissociation pathway. The sampling of the points within the elbow plots can be found in App. D, p. 165. The calculated energy barrier is 0.18 eV which is consistent with previous calculations [37]. The location of the energy barrier is in the entrance channel with a distance of the molecule from the surface  $Z_c = 1.85$  and a bond length  $d = 0.79 \text{ \AA}$ . On the clean Pd(100) surface this pathway was non-activated and the dissociation proceeds with a continuous gain of energy without any hampering barrier [29]. The formation of energy barriers after adsorption of a potassium adlayer is consistent with the experimentally observed decrease of the initial sticking probability [34]. It can also explain the observed increase of the peak temperature of hydrogen desorption in the TDS experiment [34], since the formation of an energy barrier results in an increase of the energy of activation of associative hydrogen desorption. The  $\mathbf{b}_1\text{-}\mathbf{h}_1\text{-}\mathbf{b}_1$  geometry corresponds to the minimum barrier pathway. This adsorption geometry is the site furthest away from the potassium atoms, hence one can expect that there is no direct interaction between the hydrogen and the potassium atoms. On the sulphur covered Pd(100) surface the same configuration associated with the minimum pathway has been obtained, but with a smaller barrier of 0.1 eV [33]. On the clean surface the minimum pathway was located at the palladium bridge site [29].

Figure 10.2(b) displays the PES of the  $\mathbf{t}_{\text{Pd}}\text{-}\mathbf{h}_1\text{-}\mathbf{t}_{\text{Pd}}$  pathway. An activation barrier of 0.19 eV builds up in the entrance channel at  $(d, Z_c) = (0.79 \text{ \AA}, 1.85 \text{ \AA})$ . On the sulphur covered Pd(100) surface a barrier of 0.13 eV has been calculated [33]. On adsorption of either co-adsorbate - potassium or sulphur - the energy with the molecule above the  $\mathbf{h}_1$  position with the atoms pointing towards the surface Pd atoms is slightly higher than in the  $\mathbf{b}_1\text{-}\mathbf{h}_1\text{-}\mathbf{b}_1$  adsorption configuration.

Figure 10.2(c) shows the PES for the geometry with the hydrogen molecule adsorbed at the  $\mathbf{h}_1$  site in an upright position. As it is evident from the contour plot, this pathway is repulsive as it is the case on the  $\text{S}(2 \times 2)/\text{Pd}(100)$  surface [33]. There is a shallow

local minimum at  $(d, Z_c) = (1.06 \text{ \AA}, 0.6 \text{ \AA})$  with an energy of 0.26 eV. On the clean surface there was a local minimum with  $-0.03 \text{ eV}$ . The hydrogen atoms above the  $h_1$  position are at least  $4 \text{ \AA}$  away from the co-adsorbed potassium atoms. Therefore, the increase of the energy with respect to the clean surface should not be caused by a direct interaction between the H and K atoms, but rather due to a potassium-induced modification of the local electronic structure at this site, which has been verified by Wilke and Cohen [37].

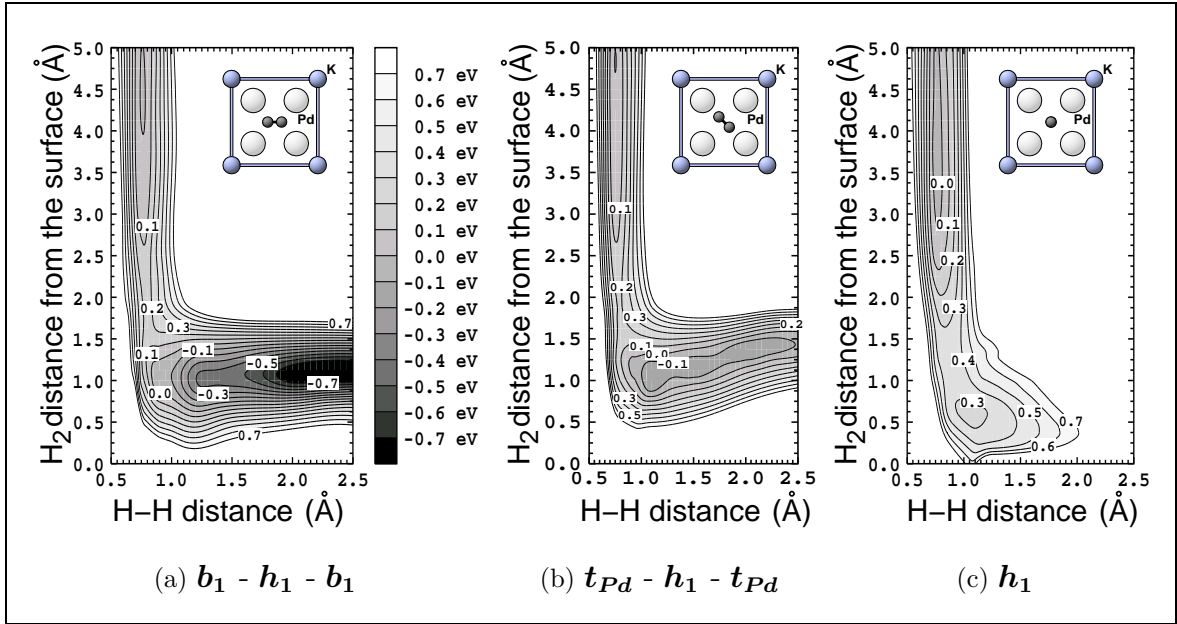


Figure 10.2: 2D cuts through the six-dimensional PES of hydrogen dissociation over  $(2 \times 2)K/Pd(100)$  above the hollow site  $h_1$ . The energy depends on the distance of the two hydrogen atoms and the distance of the molecule from the surface. The other coordinates are kept fixed. The spacing of the energy contours is 0.1 eV.

The PES for the adsorption above the hollow site  $h_2$  is illustrated in Figure 10.3. In the cut for the  $b_2-h_2-b_2$  pathway in Figure 10.3(a) there is a high barrier towards dissociation of 0.46 eV in the entrance channel. The high barrier in comparison to the  $h_1$  site is due to the smaller distance between the potassium adatoms and the hydrogen atoms. So far we have noticed that the adsorption of sulphur and potassium induces similar effects. With both adsorbates energy barriers build up in the entrance channels. The minimum energy barrier has been found to be higher on the K-covered (0.18 eV) than on the S-covered surface (0.11 eV). The height of the energy barrier depends on the distance of the hydrogen atoms from the adsorbate.

However, the  $b_2-h_2-b_2$  pathway is not purely repulsive. There is a minimum of  $E_{pot} = -0.51 \text{ eV}$  at  $(d, Z_c) = (1.87 \text{ \AA}, 1.0 \text{ \AA})$ , which is not present on the sulphur covered surface [33]. The direct repulsion between S and H atoms co-adsorbed on Pd(100) seems to be stronger than the repulsion between K and H atoms. But one has to keep in mind that sulphur adsorbs much closer to the surface ( $Z_c = 1.24 \text{ \AA}$ ) [33] than potassium ( $Z_c = 2.41 \text{ \AA}$ ), i.e. the height of the minimum in the  $b_2-h_2-b_2$  pathway is close to the sulphur adsorption height. In this configuration the H-atoms on the sulphur

covered surface are roughly 1 Å closer to the co-adsorbate than on the potassium surface leading to the observed difference.

If we now turn the  $H_2$  bond around by ninety degrees with the centre of mass still above the  $h_2$  site into the  $b_1-h_2-b_1$  geometry, the energy barrier in Fig. 10.3(b) decreases to a value of 0.27 eV at  $(d, Z_c) = (0.795 \text{ Å}, 1.85 \text{ Å})$ . The hydrogen atoms dissociate into the fourfold hollow site  $h_1$  with an energy of  $E = -0.86 \text{ eV}$  at  $(d, Z_c) = (1.1 \text{ Å}, 2.5 \text{ Å})$ . For the other  $b-h-b$  adsorption geometry above the hollow site  $h_1$  in Fig. 10.2(a) a similar value has been calculated in the exit channel, where the molecule is already dissociated. On the clean surface the corresponding energy is only  $-0.4 \text{ eV}$  [29]. This illustrates the opposite effect which potassium adatoms have during the early and late stages of the dissociative adsorption of hydrogen. They hinder the approach of the  $H_2$  molecule to the surface by inducing energy barriers in the entrance channel. In the exit channel where the molecule is dissociated, K promotes the bonding of H to the metal surface. This differs to the influence S adsorbates have on the adsorption properties of hydrogen. The presence of S adatoms decreases the adsorption energy of co-adsorbed hydrogen atoms [29, 30, 33].

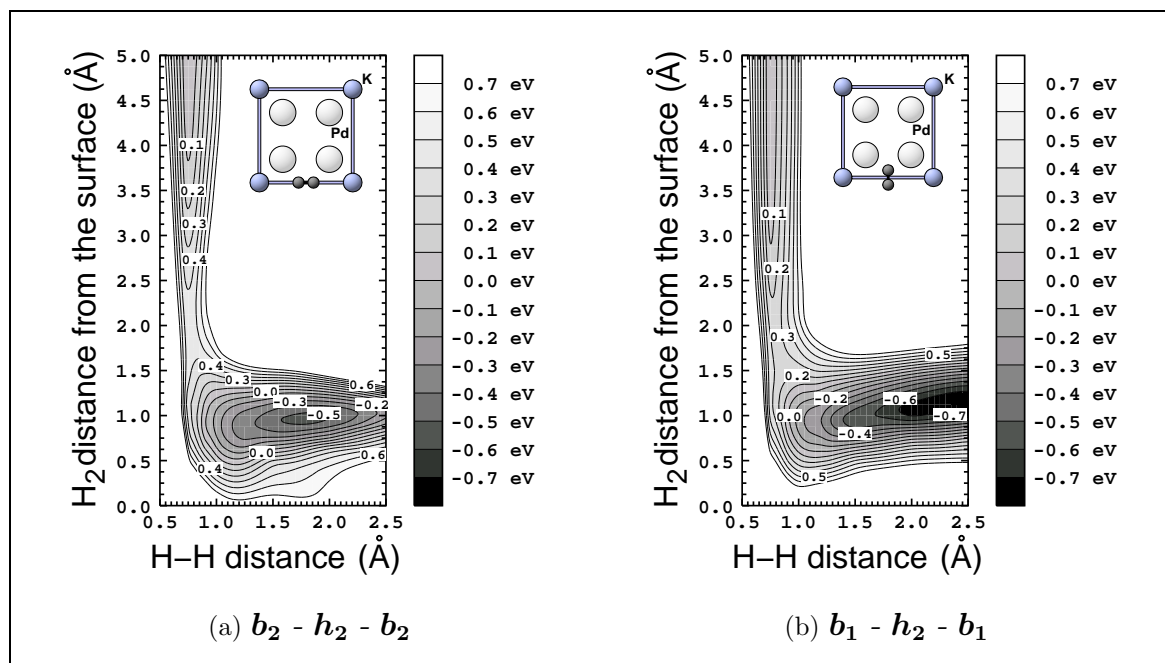


Figure 10.3: Two-dimensional cuts through the six-dimensional PES of hydrogen dissociation over  $(2 \times 2)K/Pd(100)$  above the hollow site  $h_2$ .

Wilke and Cohen reported in their previous study of the K-covered Pd(100) surface, that the energy along the reaction pathway for  $H_2$  dissociation over the two different hollow sites  $b_1-h_1-b_1$  and  $b_1-h_2-b_1$  differ only marginally [37]. This was a surprising result, because one would expect that the poisoning depends on the distance of the H atoms from the co-adsorbate as it is the case on the S-covered Pd(100) surface [29, 30, 33]. In contrary, our calculations revealed, that indeed the poisoning effect is a function of the K-H distance. The barrier in the  $b_1-h_2-b_1$  pathway was found to be 0.09 eV higher



than in  $\mathbf{b}_1\text{-}\mathbf{h}_1\text{-}\mathbf{b}_1$  geometry. As can be seen from Table 10.1 this variance is mainly due to a too large lattice constant in the previous calculation. For the reported relativistic calculation in this work we determined a bulk lattice constant of 3.95 Å, which is 1.5% larger than the experimental value. The value obtained by Wilke [22] with a non-relativistic calculation was 3.5% larger than the experimental one. The corresponding energy with a cutoff of 12 Ry in the first line of Tab. 10.1 is 0.19 eV. If we perform the same calculation but with the smaller lattice constant reported here, the energy in the second line of Tab. 10.1 rises to a value of 0.25 eV. Increasing the cutoff in the plane wave expansion to 13 Ry used in this work and performing a relativistic calculation leads to the energy barrier of 0.27 eV as printed in the last line of the Table.

The degeneracy of the energy barriers over the hollow sites found in the previous study are therefore an artefact of the extended lattice constant due to the non-relativistic calculation. The very localised Pd d-states require a relativistic treatment leading to a lattice constant, which is smaller than the non-relativistic result and agrees better with the experimental value. This results in a shorter K-H distance in the  $\mathbf{b}_1\text{-}\mathbf{h}_2\text{-}\mathbf{b}_1$  configuration and to a higher energy barrier. At the same time the energy over the other hollow site configuration  $\mathbf{b}_1\text{-}\mathbf{h}_1\text{-}\mathbf{b}_1$  is not influenced by these changes. As in the non-relativistic calculation the energy barrier was found to be 0.18 eV. This adsorption site is simply too far away from the K-adatoms to be influenced by the shorter distance.

$E_{cut}$	Lattice constant	non-relativistic	relativistic	Energy
<b>12 Ry</b>	<b>4.03 Å</b>	•		<b>0.19 eV</b>
12 Ry	3.95 Å	•		0.25 eV
12 Ry	3.95 Å		•	0.26 eV
<b>13 Ry</b>	<b>3.95 Å</b>		•	<b>0.27 eV</b>

Table 10.1: Dependence of the barrier height for the  $\mathbf{b}_1\text{-}\mathbf{h}_2\text{-}\mathbf{b}_1$  geometry on the cutoff energy, the lattice constant and the method of calculation. The first line corresponds to the result reported in Ref. [37], the last line is the result reported in this work. The experimental lattice constant is 3.89 Å. All calculations are based on the GGA [91].

## II. Two-dimensional cuts through the PES at bridge sites

We will now present the *ab initio* results of geometries, where the centre of mass of the molecule is situated at the bridge site  $\mathbf{b}_1$  of the surface unit cell. In Fig. 10.4(a) the PES for the  $\mathbf{h}_2\text{-}\mathbf{b}_1\text{-}\mathbf{h}_1$  pathway is presented. At the clean surface the dissociative adsorption in this configuration is non-activated, i.e. there is no energy barrier for dissociation present. On adsorption of a potassium adlayer a barrier of 0.19 eV builds up at a distance of 2.0 Å of the centre of mass of the molecule from the surface with a hydrogen bond length of 0.78 Å. The value of the barrier is slightly higher than the one associated with the  $\mathbf{b}_1\text{-}\mathbf{h}_1\text{-}\mathbf{b}_1$  configuration and smaller than the barrier of the  $\mathbf{b}_1\text{-}\mathbf{h}_2\text{-}\mathbf{b}_1$  geometry. The corresponding barrier on the S-covered Pd(100) surface is 0.16 eV.

The PES for the  $t_{Pd}-b_1-t_{Pd}$  geometry is plotted in Fig. 10.4(b). The dissociation process in this configuration after the co-adsorption of potassium atoms is repulsive. We obtained the same barrier of 0.19 eV as in the  $h_2-b_1-h_1$  configuration, but now shifted slightly further away from the surface at  $Z_c = 2.3 \text{ \AA}$ . There is also a shallow local minimum of 0.07 eV at  $(d, Z_c) = (1.0 \text{ \AA}, 1.7 \text{ \AA})$  present, whereas on the clean surface a local energy minimum of  $-0.2 \text{ eV}$  exists. This is due to the fact that the hydrogen atoms dissociate towards the palladium and close to the potassium atoms. The direct interaction of hydrogen and potassium leads to the increase of the energy in the shallow minimum. If the bond length stretches further, the energy rises monotonically.

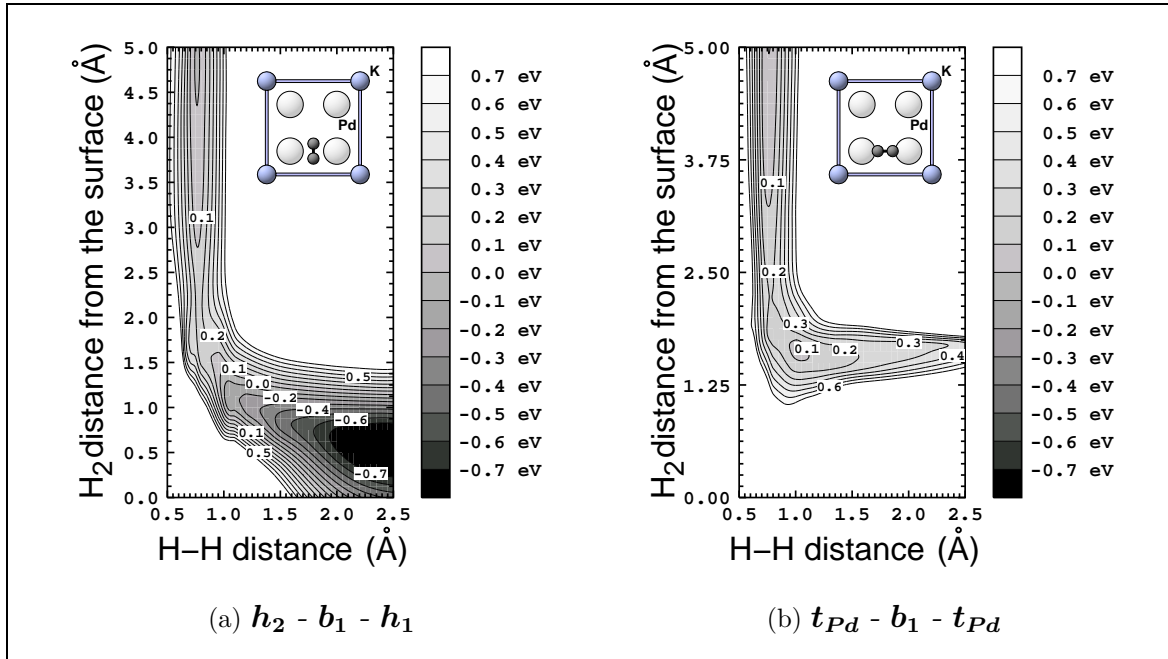


Figure 10.4: Two-dimensional cuts through the six-dimensional PES of hydrogen dissociation over  $(2 \times 2)K/Pd(100)$  above the bridge site  $b_1$ .

### III. Two-dimensional cuts through the PES at top sites

In addition to the dissociation over the hollow and bridge sites, we also considered configuration of the hydrogen molecule above top sites, i.e. on top of a palladium or a potassium atom. The potential-energy surfaces associated with on-top geometries in Fig. 10.5 differ significantly from the previously discussed hollow and bridge sites. On the clean surface the dissociation pathway  $h_2-t_{Pd}-h_2$  has a local minimum of  $-0.24 \text{ eV}$  in the entrance channel and an energy barrier of  $0.15 \text{ eV}$  in the exit channel [22]. From Fig. 10.5(a) it is visible, that due to the presence of potassium atoms the local minimum disappears and the energy barrier rises to  $0.85 \text{ eV}$  in the exit channel at  $Z_c = 2.3 \text{ \AA}$ . Such a situation can be expected, since the adsorbed potassium atom is about  $2 \text{ \AA}$  away from the adsorption site  $t_{Pd}$  and  $2.41 \text{ \AA}$  above the topmost palladium layer. The hydrogen molecule will therefore interact directly with the K atom before it reaches the palladium surface. The repulsive interaction between potassium and

hydrogen then raises the energy of the PES. A similar situation has been found for the sulphur-covered surface. However, since sulphur adsorbs only 1.24 Å above the surface, the energy barrier of 1.28 eV is located at a shorter distance from the surface ( $Z_c=1.4$  Å).

In Figure 10.5(b) the molecule approaches the surface palladium atom with its axis perpendicular to the surface. The interaction is purely repulsive, no dissociation is possible. At a distance of 2 Å from the surface the energy has increased to a value of 1.1 eV. The repulsion is even stronger if the upright molecule approaches the potassium co-adsorbate in Fig. 10.5(c). Far away from the topmost palladium layer at  $Z_c=4.3$  eV the energy reaches a value of 1.8 eV. These findings support the observation that an upright hydrogen molecule is not able to dissociate as it has been reported already in the studies concentrated on the clean and the sulphur-covered surface [22, 29, 30, 33].

Figure 10.5(d) shows the PES over a surface potassium atom within the  $b_2-t_K-b_2$  configuration. In this geometry the molecule approaches the potassium atom directly which leads to a strong repulsion. On the sulphur-covered surface the corresponding dissociation process was activated with a barrier of 2.55 eV. However, with potassium adsorbates the process is purely repulsive. We calculated energies up to 5 eV without finding a saddle point in the energy landscape. The potassium atoms adsorb far away on the surface. Although dissociation in this configuration corresponds to a motion of the hydrogen atoms away from the potassium atom which usually leads to an energy gain, the distance to the surface is too large to experience the attraction by the metal atoms.

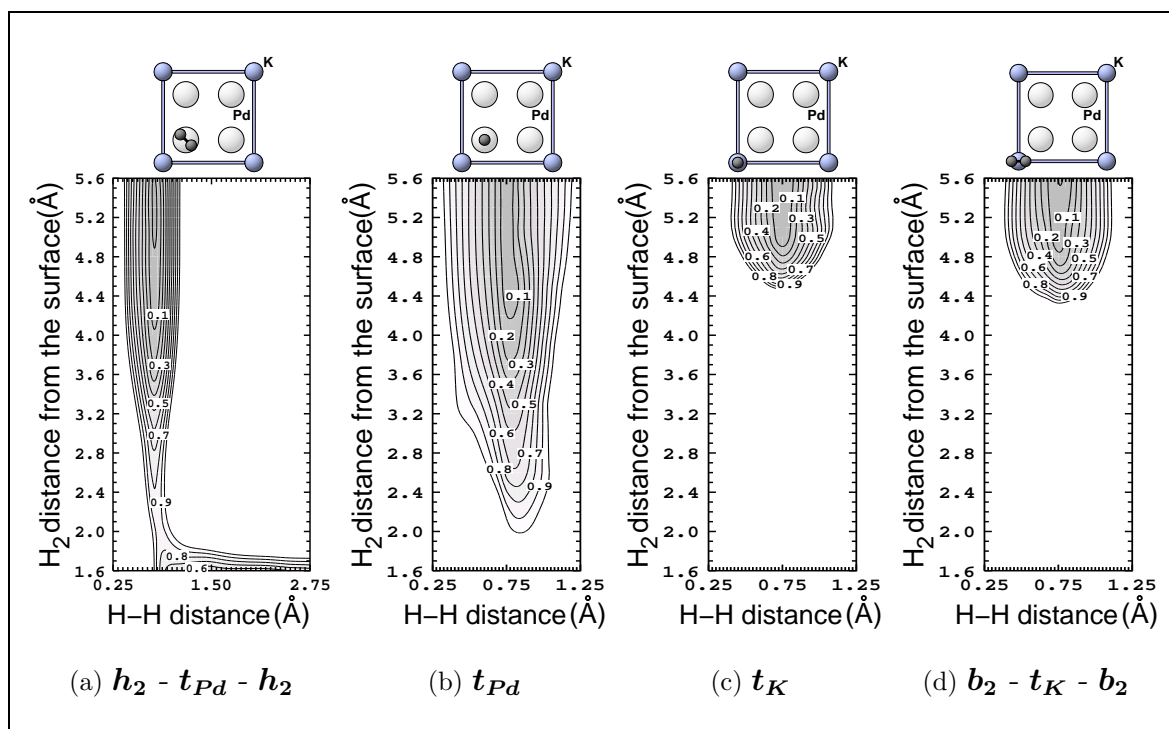


Figure 10.5: 2D cuts through the 6D PES of  $H_2$  dissociation over  $(2 \times 2)K/Pd(100)$  above the palladium top site  $t_{Pd}$  in (a)+(b) and the potassium top site  $t_K$  in (c)+(d).

#### IV. Corrugation of the PES

In addition to the calculation of two dimensional cuts above high-symmetric sites we calculated the corrugation of the energy barriers as a function of the lateral coordinates for different fixed distances and bond lengths of the molecule from the surface. The most interesting configuration corresponds to the situation when the molecule is oriented parallel to the surface in the bond breaking region. In such a geometry hydrogen dissociates on the Pd(100) surface. For  $H_2$  above or between hollow and bridge sites the energy barriers are located at a bond length of  $d=0.77-0.79 \text{ \AA}$  and a distance of the centre of mass from the surface between  $1.8 \text{ \AA}$  and  $2.0 \text{ \AA}$ .

Figure 10.6 shows the corrugation of the energy barriers for the molecule moved from the  $b_1-h_2-b_1$  geometry, over the  $b_1$  site, to the  $b_1-h_1-b_1$  configuration. The polar and azimuthal angles are fixed at a value of  $90^\circ$ . The H-H distance is  $d=0.79 \text{ \AA}$ . With the molecule above the  $h_2$  site corresponding to  $X=0$  a in Fig. 10.6, where  $a$  is the length of the  $(2 \times 2)$  cell, the energy increases while the molecule approaches the surface from a distance of  $2.0 \text{ \AA}$  to  $1.8 \text{ \AA}$ . If the molecule further moves towards the surface, the energy decreases again, which can be seen in the elbow plot in Fig 10.3(b), p. 116. The dissociation is activated. The barrier is  $0.27 \text{ eV}$  at  $Z=1.8 \text{ \AA}$ . Above the hollow site  $h_1$  ( $X=0.5$  a in Fig.10.6) the energy changes only slightly for the four different vertical distances. The maximum is located a  $Z=1.85 \text{ \AA}$  with an energy of  $0.18 \text{ eV}$ . In an intermediate configuration, e.g. above the  $b_1$  site ( $X=0.25$  a), the highest barrier is situated further away from the topmost Pd layer at  $Z=2.0 \text{ \AA}$ , i.e. the surface exhibits not only energetic but also a slight geometric corrugation. The outward shift of the barrier is present between  $X=0.15-0.35$  a and is therefore due to the short distance of the molecule to the Pd atoms. Between the bridge site  $b_1$  and the hollow site  $h_1$  there is a slight local minimum at  $X=0.3$  a, followed by a shallow maximum at  $X=0.4$  a, before

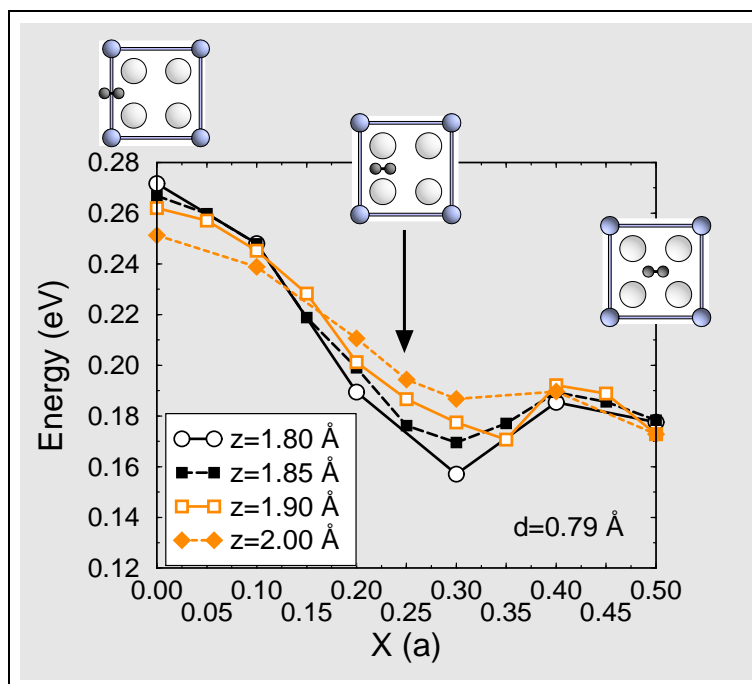
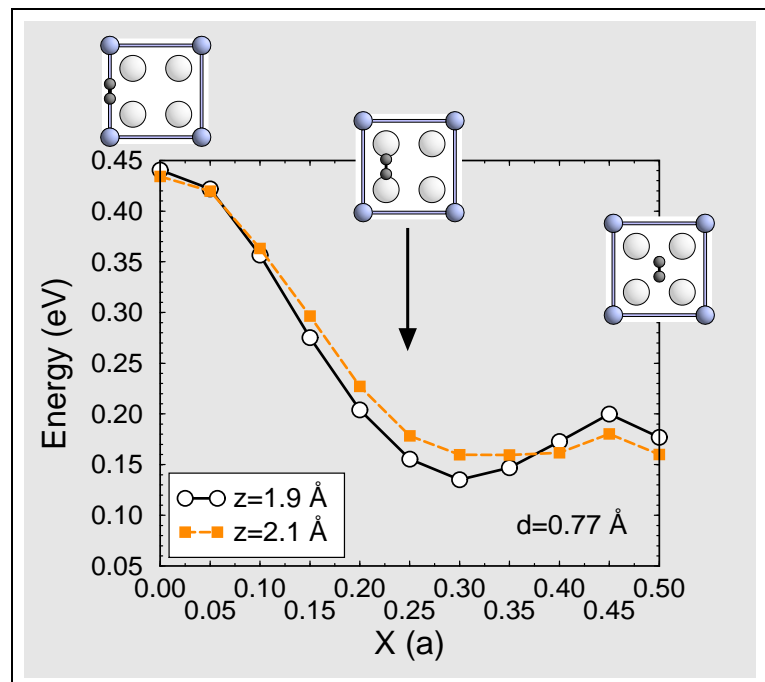


Figure 10.6: Corrugation of the energy along the  $x$ -axis of the unit cell for a hydrogen molecule parallel to the surface with  $(\theta, \phi) = (90^\circ, 90^\circ)$  at four different heights  $Z$ . The bond length of the molecule is  $d=0.79 \text{ \AA}$ . The lateral coordinate  $X$  is plotted as a function of the length of the  $(2 \times 2)$  unit cell  $a$ . The  $H_2$  molecule is moved from the  $b_1-h_2-b_1$  ( $X=0$  a) to the  $b_1-h_1-b_1$  ( $X=0.5$  a) configuration.

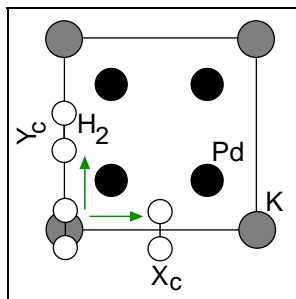
the energies drop to the value of the minimum pathway above  $\mathbf{h}_1$ . The reason for the lower barrier at  $X=0.3a$  is that one H atom gets very close to the favourable adsorption site  $\mathbf{h}_1$  and the other is further away from the repulsive K atom. However, this is a very small effect. If we focus on the outline of the different curves and concentrate on the highest barriers at each lateral coordinate  $X$ , the energy decreases monotonically from the  $\mathbf{h}_2$  to the  $\mathbf{h}_1$  site. The energetic corrugation is not very pronounced, the difference between the highest and the smallest barrier in Fig. 10.6 is just 0.09 eV. For comparison, the same difference on the sulphur-covered surface is 0.5 eV [33], i.e. the corrugation on the K(2×2) is much weaker than on the S(2×2)/Pd(100) surface.

If we turn the molecular axis of the hydrogen molecule in-line with the surface potassium atoms, the energetic corrugation is much more pronounced as shown in Figure 10.7. The molecule is again oriented parallel to the surface, but the azimuthal angle has been changed from  $90^\circ$  to  $0^\circ$ . The bond length is  $0.77 \text{ \AA}$  and the distances from the surface are  $1.9 \text{ \AA}$  and  $2.1 \text{ \AA}$ , i.e. corresponding to the region where the  $\text{H}_2$  bond starts to break. In this configuration the dissociating H atoms point directly at the repulsing surface Pd and K atoms, increasing the energy barrier. The difference between the  $\mathbf{h}_2$  on the left and the  $\mathbf{h}_1$  adsorption site on the right has been raised to 0.27 eV. Yet, on the sulphur-covered surface the energy barriers above the hollow configurations in this geometry differ by 0.65 eV, i.e. the corrugation is again significantly larger. The shape of the curves is the same as in the previous plot, the highest barriers decrease from the  $\mathbf{b}_2\text{-}\mathbf{h}_2\text{-}\mathbf{b}_2$  to the  $\mathbf{b}_1\text{-}\mathbf{h}_1\text{-}\mathbf{b}_1$  configuration. Again, close to the palladium atoms the energy barrier is shifted further away from the surface. Indeed, we found for the  $\mathbf{t}_{Pd}\text{-}\mathbf{b}_1\text{-}\mathbf{t}_{Pd}$  geometry that the energy increases by 20 meV up to a distance of  $Z = 2.3 \text{ \AA}$ , which we have not plotted here. We point out, that the shown energies do not necessarily correspond to the highest barrier above each site. The location of the barrier also depends on the bond length, which we fixed in Fig. 10.6 and Fig. 10.7.

Figure 10.7: *Corrugation of the energy along the x-axis of the unit cell for a hydrogen molecule parallel to the surface with  $(\theta, \phi) = (90^\circ, 0^\circ)$  at two different heights  $Z$ . The bond length of the molecule is  $d=0.77 \text{ \AA}$ . The lateral coordinate  $X$  is plotted as a function of the length of the  $(2 \times 2)$  unit cell  $a$ . The  $\text{H}_2$  molecule is moved from the  $\mathbf{b}_2\text{-}\mathbf{h}_2\text{-}\mathbf{b}_2$  ( $X=0a$ ) to the  $\mathbf{b}_1\text{-}\mathbf{h}_1\text{-}\mathbf{b}_1$  ( $X=0.5a$ ) configuration.*



Additionally, we have also calculated the variation of the barrier height as a function of both lateral coordinates  $X_c$  and  $Y_c$ . We kept the molecule in the same position as before but now moved the centre of mass across the unit cell and not only along the line between the  $h_1$  and the  $h_2$  hollow site. The molecule is fixed at a distance from the surface  $Z_c = 1.9 \text{ \AA}$  with a bond length of  $d = 0.77 \text{ \AA}$  at angles  $(\theta, \phi) = (90^\circ, 0^\circ)$  and the lateral coordinates vary between  $0a$  and  $0.5a$ , where  $a$  is the length of the  $(2 \times 2)$  cell. Examples of configurations of the hydrogen molecule are plotted in Fig. 10.8.



**Figure 10.8:**  $H_2$  geometries for Fig. 10.9

The *ab initio* total energy as a function of the two lateral coordinates in the above configuration can be seen in the contour plot of Fig. 10.9. The calculated grid points are indicated by the white squares. We have determined energies up to  $3.3 \text{ eV}$  at  $(X_c, Y_c) = (0.2a, 0.2a)$ . Starting from the hollow site  $h_1$  in the middle of Fig. 10.9 the energy barrier increases rather slowly towards the palladium on-top site followed by a steep increase to the potassium on-top site. A quarter of the area of the unit cell exhibits an energy barrier smaller than  $0.2 \text{ eV}$ , reaching a value of  $43\%$  for an energy of  $0.4 \text{ eV}$ . Hence, a rather larger area of the surface is accessible for sticking in the energy range we are aiming at ( $\leq 1 \text{ eV}$ ), even if the steering effect would not be present.

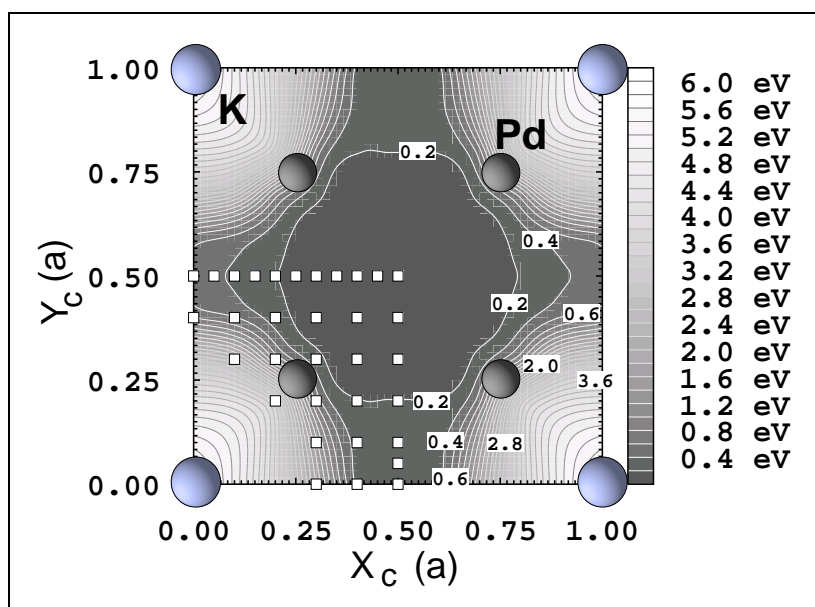


Figure 10.9: Potential energy as a function of the lateral coordinates  $X_c$  and  $Y_c$ . The molecule is oriented parallel to the surface at a height  $z = 1.9 \text{ \AA}$  with a bond length  $d = 0.77 \text{ \AA}$  as plotted in Fig 10.8. The contour spacing is  $0.2 \text{ eV}$ . The white squares mark the calculated *ab initio* points.  $a$ : length of the  $(2 \times 2)$  cell.

This also sheds a new light on the discussion of the corrugation of the sulphur covered  $\text{Pd}(100)$  surface in Sec. 9.5, p. 104. For this surface, only the high symmetric configurations have been determined by *ab initio* methods, whereas the rest of the corrugation was analytically interpolated. The analytical corrugation “assumed” first a slow and then a steep increase from the hollow sites towards the top-sites, which is on the potassium covered surface indeed the case. Hence, the discrepancy between the analytical and the experimental sticking curve can not be solely due to the form of the analytical corrugation, but probably rather - as proposed - to the presence of

sub-surface sulphur in the sample [51]. However, we should keep in mind that the corrugation on the sulphur covered sample is much more pronounced than on the potassium surface except for the vicinity of the K or S atoms, respectively. Therefore, the form of the corrugation can still have an influence on the observed discrepancy and it will be interesting to calculate this dependency in the future.

## V. Summary of the *ab initio* results

In conclusion, using density-functional theory and the FP-LAPW method we have investigated the effect of potassium adatoms on hydrogen dissociation over the  $(2 \times 2)\text{K}/\text{Pd}(100)$  surface over the high-symmetric sites  $\mathbf{h}_1$ ,  $\mathbf{h}_2$ ,  $\mathbf{b}_1$ ,  $\mathbf{t}_{\text{Pd}}$  and  $\mathbf{t}_{\text{K}}$ . We found that the dissociation behaviour depends on the distance between the hydrogen centre of mass and the adsorbed potassium atoms, and on the orientation of the dissociating molecule. Potassium adatoms hinder hydrogen dissociation by forming energy barriers in the entrance channel. Still, the dissociative adsorption is exothermic, the hydrogen molecule gains energy during the reaction. The minimum pathway has been determined to be over the surface hollow site  $\mathbf{h}_1$  with the hydrogen atoms oriented towards the palladium bridge sites. The energy barrier in this configuration is 0.18 eV. In the exit channel, when the molecules are dissociated, the presence of potassium adatoms tends to stabilise the adsorption of the single hydrogen atoms, which is in contrast to the sulphur-covered Pd(100) surface [29,30,33].

The results can be summarised as follows:

- For an upright hydrogen molecule, i.e. with its molecular axis perpendicular to the surface, the PES is repulsive above all sites ( $\mathbf{h}_1$ ,  $\mathbf{t}_{\text{Pd}}$ ,  $\mathbf{t}_{\text{K}}$ ).
- For adsorption over the hollow site  $\mathbf{h}_1$  with the molecular axis parallel to the surface, the PES is activated ( $\mathbf{b}_1\text{-}\mathbf{h}_1\text{-}\mathbf{b}_1$  and  $\mathbf{t}_{\text{Pd}}\text{-}\mathbf{h}_1\text{-}\mathbf{t}_{\text{Pd}}$  pathways). The barrier is located in the entrance channel with an energy of about 0.18 eV resulting from a potassium-induced modification of the local electronic structure at the surface. In these geometries the molecule reaches the surface more than 3.5 Å away from the K atoms.
- For shorter distances, i.e. if the distance between the adsorption site and the potassium atom is between 1.5 Å and 3. Å, with the molecule still oriented parallel to the surface but not approaching the palladium atoms directly, the dissociation is also activated. The height of the barrier depends on the distance of the hydrogens from the K atoms. The energy barrier is in the entrance channel resulting from a direct interaction of hydrogen with the co-adsorbate. The corresponding adsorption configurations are  $\mathbf{b}_1\text{-}\mathbf{h}_2\text{-}\mathbf{b}_1$ ,  $\mathbf{h}_2\text{-}\mathbf{b}_1\text{-}\mathbf{h}_1$ ,  $\mathbf{b}_2\text{-}\mathbf{h}_2\text{-}\mathbf{b}_2$  and  $\mathbf{h}_2\text{-}\mathbf{t}_{\text{Pd}}\text{-}\mathbf{h}_2$ .
- If the dissociating hydrogen atoms approach the palladium atoms directly, the PES is repulsive ( $\mathbf{t}_{\text{Pd}}\text{-}\mathbf{b}_1\text{-}\mathbf{t}_{\text{Pd}}$ ). Furthermore, if the adsorption site is close to a potassium atom the PES is also repulsive ( $\mathbf{b}_2\text{-}\mathbf{t}_{\text{K}}\text{-}\mathbf{b}_2$ ).

In Table 10.2 we compare the results of the potassium-covered Pd(100) surface in eight different configurations with sulphur as an adsorbate [33]. On both surfaces the

minimum barrier can be found above the hollow site  $h_1$ . However, the presence of potassium leads to a higher minimum barrier of 0.18 eV in comparison to 0.1 eV with sulphur atoms. Furthermore, the corrugation of the energy barriers at the  $S(2 \times 2)$  surface is larger than on the  $K(2 \times 2)$ , which can be seen from the value of 0.6 eV above the  $h_2$  site for the former system and 0.27 eV for the latter.

Another difference is the adsorption above the hollow site  $h_2$  with the H atoms pointing at the adatoms. If the adsorbate is sulphur, then the PES is repulsive in this configuration. If it is potassium, the process is activated with a barrier of 0.46 eV. This variance is due to the different adsorption heights of sulphur and potassium on the Pd(100) surface. Sulphur binds to the surface more closer, at a height of 1.24 Å, whereas potassium is more repulsive and adsorbs 1.2 Å further away. The potential energy for the dissociated hydrogen atoms has its minimum at a distance of about 1 Å to the surface, i.e. close to the adsorption height of sulphur. This leads to a repulsive PES on the  $S(2 \times 2)$  surface, whereas on the  $K(2 \times 2)$  the K-H distance is still large enough to allow dissociation.

On the other hand, on adsorption of sulphur dissociation over the on-top site with the  $H_2$  axis parallel to the surface ( $t_S$ ,  $\theta = 90^\circ$ ,  $\phi = 0^\circ$ ) is still possible, though the process is activated with a large barrier of 2.55 eV. Over a potassium atom the PES is purely repulsive. The reason is also related to the adsorption height. Since potassium is further away than sulphur from the topmost palladium layer, the hydrogen molecule is not able to experience the attraction of the metal surface for dissociation but only the repulsion by the adatom.

Site	$H_2$ geometry ( $\theta, \phi$ )	$S(2 \times 2)/Pd(100)$		$K(2 \times 2)/Pd(100)$	
		Energy barrier	Process	Energy barrier	Process
$h_1$	( $90^\circ, 90^\circ$ )	$E_b = 0.10$ eV	activated	$E_b = 0.18$ eV	activated
$h_2$	( $90^\circ, 90^\circ$ )	$E_b = 0.60$ eV	activated	$E_b = 0.27$ eV	activated
$h_2$	( $90^\circ, 0^\circ$ )	$E_b \geq 0.75$ eV	repulsive	$E_b = 0.46$ eV	activated
$h_1$	( $90^\circ, 135^\circ$ )	$E_b = 0.13$ eV	activated	$E_b = 0.19$ eV	activated
$h_1$	( $0^\circ, 0^\circ$ )	$E_b \geq 0.25$ eV	repulsive	$E_b \geq 0.34$ eV	repulsive
$b_1$	( $90^\circ, 90^\circ$ )	$E_b = 0.15$ eV	activated	$E_b = 0.19$ eV	activated
$t_{Pd}$	( $90^\circ, 135^\circ$ )	$E_b = 1.28$ eV	activated	$E_b = 0.85$ eV	activated
$t_S, t_K$	( $90^\circ, 0^\circ$ )	$E_b = 2.55$ eV	activated	$E_b \geq 5.00$ eV	repulsive

Table 10.2: Comparison of barrier heights for the hydrogen dissociation over the  $(2 \times 2)$  sulphur and potassium covered Pd(100) surface at different adsorption sites and geometries. The energies are given per  $H_2$  molecule. The results for the sulphur covered surface have been obtained from Ref. [33].



In order to understand the poisoning effect of an adsorbate on a surface different models have been introduced. For a discussion of the reactivity of clean transition metal surfaces towards hydrogen dissociation we refer to, e.g., Refs. [20, 36]. For instance, Wilke and Cohen compared the changes of the density of states (DOS) induced by a potassium and a sulphur adsorbate above the  $h_1$  site [22]. Wei, Groß and Scheffler extended this work for the sulphur covered surface and studied the DOS in more detail [33]. We will briefly repeat the results of both works. Wilke and Cohen focused on the most sensitive region of the PES just before the end of the entrance channel, where the top of the dissociation barrier of hydrogen dissociation on these surfaces is situated. They found that both types of adsorbates modify the surface electronic structure by downshifting and broadening surface Pd d-states. The interaction of hydrogen with the broadened band of Pd d-states leads to the occupation of H<sub>2</sub>-substrate antibonding states. Thus, a repulsive contribution of the hydrogen-surface interaction appears and gives rise to the formation of energy barriers. Closer to the adatom a direct interaction leads to a large energy barrier [33]. The origin of these modifications of the surface density of states, however, is completely different in the two cases of K or S adsorbates. Sulphur forms strong covalent bonds with the surface, whereas the interaction of potassium with the surface has a strong ionic component [37].

In the following sections we will now discuss how the *ab initio* calculations can be interpolated with neural networks and what the dynamical consequences of the poisoning effect of potassium adsorbates on a Pd(100) surface in comparison to the sulphur-covered sample are.

### 10.3 Incorporation of the symmetry

In order to *a priori* represent the symmetry within neural networks we used the same terms as on the sulphur-covered Pd(100) surface. But instead of using reaction path coordinates we apply the more natural relative and centre of mass coordinates. Furthermore, we employ a weighting factor of  $e^{(-Z_c/2)}$ , where  $Z_c$  is the distance of the centre of mass of the molecule from the surface, in order to ensure, that the energy in the vacuum only depends on the bond length. The new set of eight coordinates, i.e. the inputs to the neural network, are:

$$X_1 = d,$$

$$X_2 = e^{(-Z_c/2)},$$

$$X_3 = \sin^2(\theta) \cos(2\phi) [\cos(G \cdot X_c) - \cos(G \cdot Y_c)] e^{(-Z_c/2)},$$

$$X_4 = \sin^2(\theta) \cos(2\phi) [\cos(2G \cdot X_c) - \cos(2G \cdot Y_c)] e^{(-Z_c/2)},$$

$$X_5 = \cos^2(\theta) e^{(-Z_c/2)},$$

$$X_6 = [\cos(G \cdot X_c) + \cos(G \cdot Y_c)] e^{(-Z_c/2)},$$

$$X_7 = [\cos(2G \cdot X_c) + \cos(2G \cdot Y_c)] e^{(-Z_c/2)},$$

$$X_8 = \sin^4(\theta) \cos(4\phi) [\cos(2G \cdot X_c) + \cos(2G \cdot Y_c)] e^{(-Z_c/2)}.$$

## 10.4 Neural Network PES

To get a continuous representation of the potential-energy surface the *ab initio* data have been interpolated with neural networks. We divided the 659 *ab initio* energies into a training set of 619 points and a test set containing 40 energies. In addition, 60 energies representing the vacuum region of the molecule have been used in the training set by repeating the *ab initio* calculated energy at  $Z = 5.05 \text{ \AA}$  to distances of up to  $Z = 15 \text{ \AA}$ . A  $8 - 24 - 18 - 1$  sl neural network, i.e. with two hidden layers and 685 weights, has been applied for the interpolation. The Kalman Filter parameters were  $\lambda(0) = 0.98$ ,  $\lambda_0 = 0.99903$  and the adaptive threshold parameter was 0.1. During the minimisation of the error function the data corresponding to the valley of each elbow plot have been weighted five times and the region of the energy barrier 20 times higher than the rest of the energies. The training and test root mean squared error as a function of the number of presentations of the whole data set (epoch) to the neural network is shown in Fig. 10.10. The training and test error drop very fast and start to saturate at around 100 epochs. Both error functions continue to go down with longer training time, but after 100 epochs the test error decreases only by further 30 meV. The training error after 5000 epochs measured 21 meV, the test error on the 40 energies not presented during the optimisation of the network weights was 74 meV. Both errors lie well below the desired accuracy of 0.1 eV. The training time for the whole 5000 epochs was nine hours on a IBM-SP2 machine. The first important 100 epochs took only 10 minutes. In any case, the interpolation time is just a small fraction of the underlying density functional theory calculations.

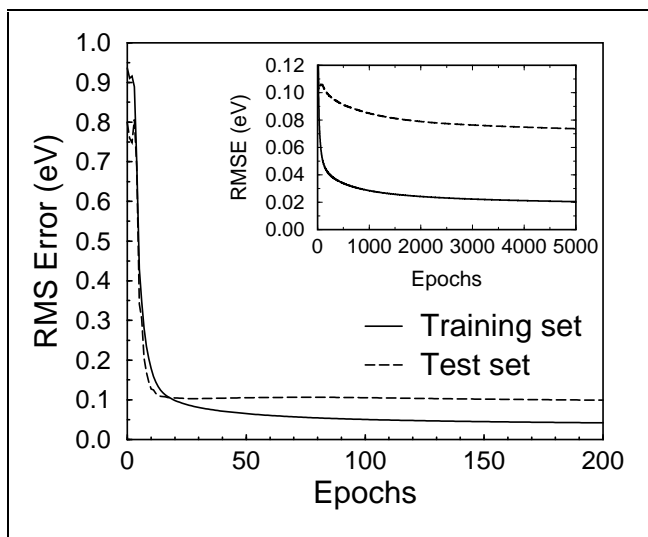


Figure 10.10: The training and test root mean squared errors as a function of the number of epochs for the neural network fit of the *ab initio* energies of the system  $H_2/K(2 \times 2)/Pd(100)$ . Inset: zoom into the error functions for smaller error values. A  $8 - 24 - 18 - 1$  sl neural network has been used to fit the training set of 679 energies. The test set consists of 40 *ab initio* energies not presented to the network during the optimisation.

Fig. 10.11 illustrates the accuracy of the obtained neural network model. It shows the distribution of the prediction error for the training and test set, respectively. 99.6% of the training data have an error smaller than 0.1 eV. Only three examples exceed this level. The highest absolute error is only 0.12 eV and occurs at an energy of 1.2 eV, i.e. in a region where the potential is already elevated. This region has been fitted with a smaller weight during the minimisation of the error function. It cannot be reached in

our molecular dynamics calculations, which we perform for initial kinetic energies of the molecule of up to 0.8 eV.

Also the 40 *ab initio* total energies contained in the test set are well reproduced by the neural network PES. Already 30 geometries exhibit an error of 0.05 eV and less and 38 energies have an absolute error of less than 0.1 eV. Only two examples exceed the latter value. Both these geometries correspond again to the region far away from the valley of the elbow plots. The associated potential energies are 1.4 and 2.8 eV.

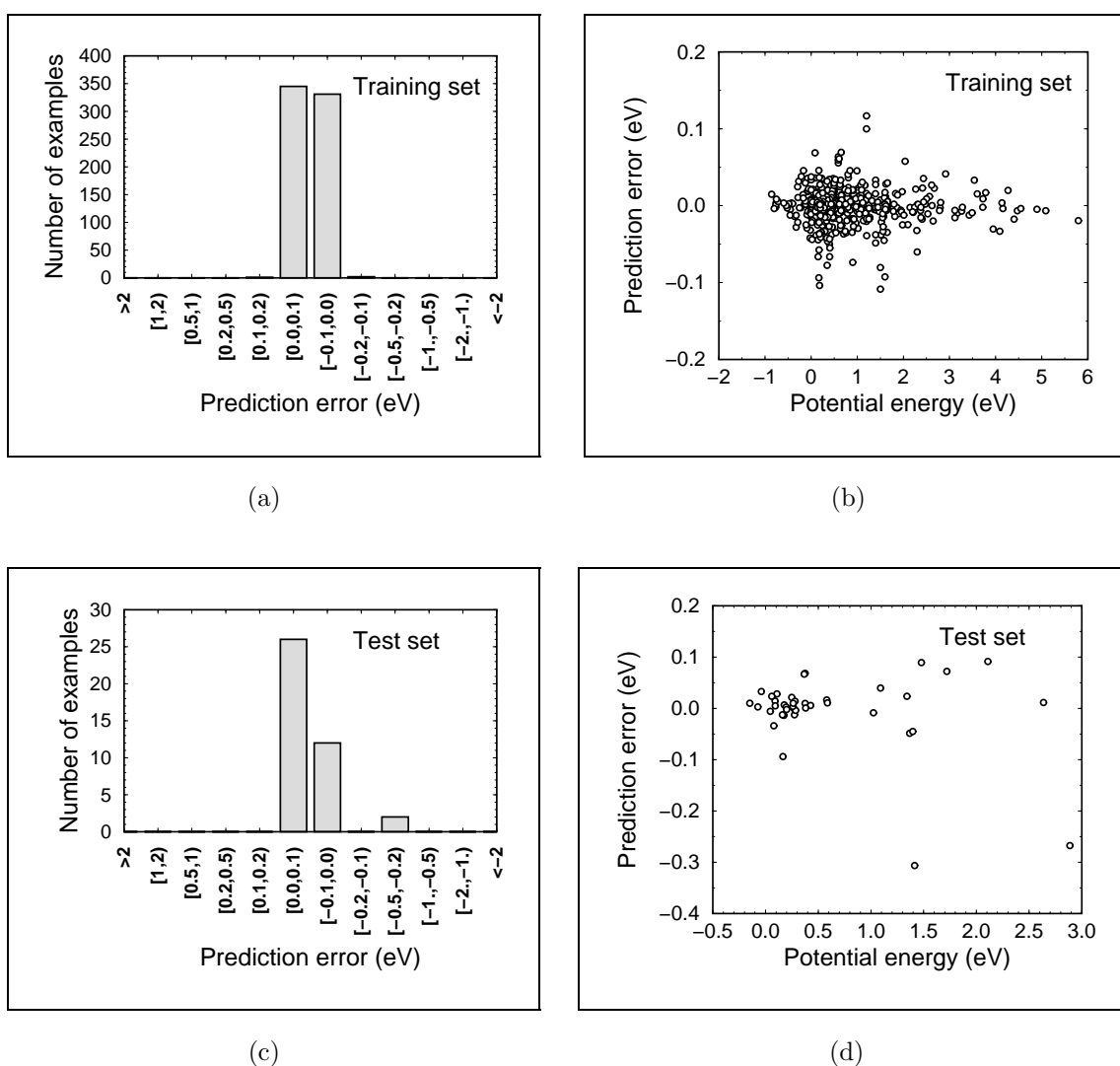


Figure 10.11: Training and test error of the neural network fit to the *ab initio* energies of the system  $H_2/K(2 \times 2)/Pd(100)$ . The training set contains 679 energies, the test set 40 energies. Fig. (a) and (c) show the number of training and test examples versus the prediction error. Fig. (b) and (d) display the prediction error of the training and test set versus the potential energy of the data. Only three training examples or 0.4% of the training data have an error higher than 0.1 eV, but still smaller than 0.2 eV. Furthermore, only two test geometries have an error higher than 0.1 eV.

In Fig. 10.12 we plotted three two-dimensional cuts through the six-dimensional neural network PES above different adsorption sites: the hollow site  $h_1$ , the hollow site  $h_2$  and above the potassium on-top site. The elbow plots obtained from the neural network PES approximate the *ab initio* cuts of section 10.2.2 very well. Not only the general shape of the cuts is well reproduced, like the repulsive character above the potassium on-top site, but also the barriers are fitted nicely. For instance, Fig. 10.12(a) corresponds to the minimum path towards dissociation with a barrier of 0.18 eV, cf. with the *ab initio* cut in Fig. 10.2(a), p. 115.

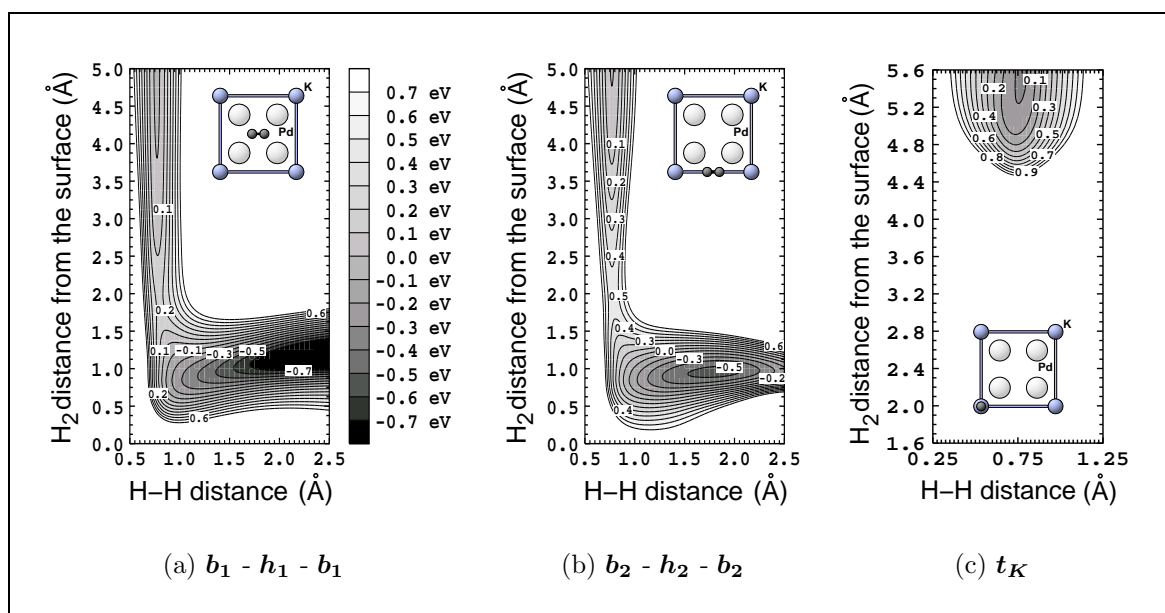


Figure 10.12: Two-dimensional cuts through the six-dimensional neural network PES for the dissociation of hydrogen on a potassium covered Pd(100) surface. The molecule approaches the surface (a) above the hollow site  $h_1$ , (b) above the hollow site  $h_2$  and (c) on top of a potassium atom. For comparison, the corresponding *ab initio* cuts are Fig. 10.2(a), p. 115, Fig. 10.3(a), p. 116 and Fig. 10.5(c), p. 119, respectively

### 10.4.1 Constrained fitting approach

The above presented fitting of *ab initio* total energies with a single neural network is general and in principle transferable to other problems. Its application is not only limited to the dissociation of diatomic molecules on a metal surface, as we will show in the last chapter of this work. Yet, sometimes it may be desirable or even necessary to include available physical knowledge - as present implicitly in the data - explicitly into the interpolation. In the discussion of the PES for the sulphur covered Pd(100) surface, we demonstrated that neural networks can also be used in a more restricted context. In the constrained fitting approach using two neural networks we explicitly secured by a constraint, that every elbow plot has a barrier towards dissociation which is at least as high as the minimum barrier (see Sec. 9.4). This was done by first fitting a two-dimensional lower bound for the energy of *all* elbow plots with one neural network.

Then, in a second step, we fitted the six-dimensional problem with a second neural network under the constraint:

$$V_{PES}^{NN}(\{b_w\}, X, Y, Z, d, \theta, \phi) = V_{2D}^{NN}(\{a_w\}, Z, d) + [V_{6D}^{NN}(\{b_w\}, X, Y, Z, d, \theta, \phi)]^2, \quad (10.1)$$

i.e. the energy in the  $Zd$  plane was allowed to be only higher than the previously fitted lower bound. Here the vector  $\mathbf{a}_w$  is the set of weights corresponding to the 2D lower bound and  $\mathbf{b}_w$  is the weight vector associated with the optimisation of the full 6D problem. We have tested this approach also for the potassium covered surface and present the results here and in the next subsection. However, we emphasise again that the constrained fitting approach requires significant insight into the problem, which is often not available. Therefore the general scheme using only one neural network for the interpolation of the full problem will be preferred.

The first step of the constrained neural network interpolation approach requires the fit of a two-dimensional lower bound of all the calculated elbow plots. Yet, the 2D cut corresponding to the minimum pathway on the potassium covered Pd(100) surface is not strictly lower in energy compared to *all* other elbow plots *for all*  $Zd$  values. Recall that we have not only energetic but also geometric corrugation, i.e. the energy barrier is not located at the same position in the different 2D cuts. We constructed an “artificial” lower energy bound from the *ab initio* data following the rule (see Sec. 9.4):

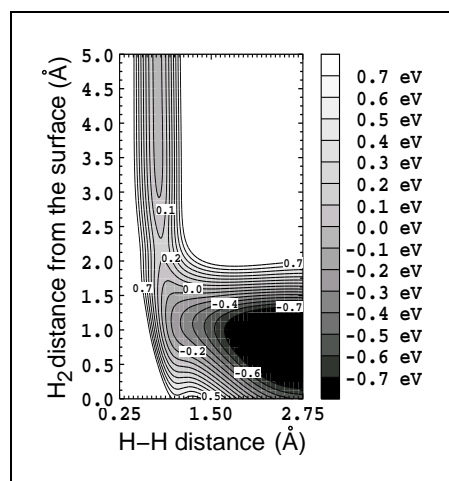
If none of the calculated elbow plots is a strictly lower bound to all other 2D cuts, one can generate this lower bound as follows:

- (1) Perform a 2D interpolation of each calculated elbow plot.
- (2) Take the minimum energy on a grid of  $Z, d$  values of *all* interpolated elbow plots. The generated set of values forms the lower bound.

The 2D lower energy bound constructed in this way is shown in Fig. 10.13 (see Sec. 9.4). For the interpolation of the gridded  $Zd$  values we used a 2–24–18–1 sl network.

In the second step we interpolated the full problem employing a 8–24–18–1 sl network and the same training and test set of *ab initio* energies as before. The root mean squared error after 500 epochs was 0.072 eV and 0.088 eV for the training and test set, respectively. The errors were slightly larger than the ones of the general fit with one neural network but still within the desired accuracy range.

We have also tested the convergence of the full problem if we use the “real” minimum pathway  $\mathbf{b}_1\text{-}\mathbf{h}_1\text{-}\mathbf{b}_1$  as the 2D lower energy bound in the first step of the constrained fitting approach. We were not able to achieve convergence to a training error lower than 0.3 eV. This reflects that the minimum pathway does not exhibit a lower energy for all  $Zd$  values compared to the rest of the elbow plots and the constraint in Eq. 10.1 is violated, as expected.



**Figure 10.13:** 2D lower energy bound of the constrained fitting approach for  $H_2/K/Pd(100)$ .

## 10.5 Molecular dynamics with neural networks

Based on the neural network PES, we have performed classical molecular dynamics simulations of the dissociative adsorption of hydrogen on the potassium covered Pd(100) surface under normal incidence and determined the sticking probability. For each energy we calculated 2000 trajectories, i.e. we followed a total number of 76000 trajectories through the configuration space. The initial geometries were drawn from a random sampling. The statistical error of the sticking probability at a given energy can be expressed as  $1/\sqrt{N}$ , where  $N$  is the number of trajectories. For a number of 1000 trajectories the statistical error is equal to 3.2%, for 2000 trajectories it is 2.2%. The sticking curve and its error bars are shown in Fig. 10.14. In addition to the neural network interpolation, A. Groß performed separately an analytical fit to the *ab initio* data. The adsorption probability based on this analytical PES with 1000 trajectories per energy is also plotted in Fig. 10.14. Both dynamical results agree well. Obviously, both methods are able to describe the dissociation PES reliably. However, we emphasise that the neural network approach to the interpolation of *ab initio* data is *general*, i.e. it is not restricted to the application of dissociation problems only. Furthermore, neural networks can easily be adjusted to higher dimensions. In contrast, the analytical interpolation is especially designed to represent six-dimensional PESs of the dissociation of a diatomic molecule and is not immediately transferable to other problems. In addition, the choice of basis functions with increasing dimensions is very cumbersome.

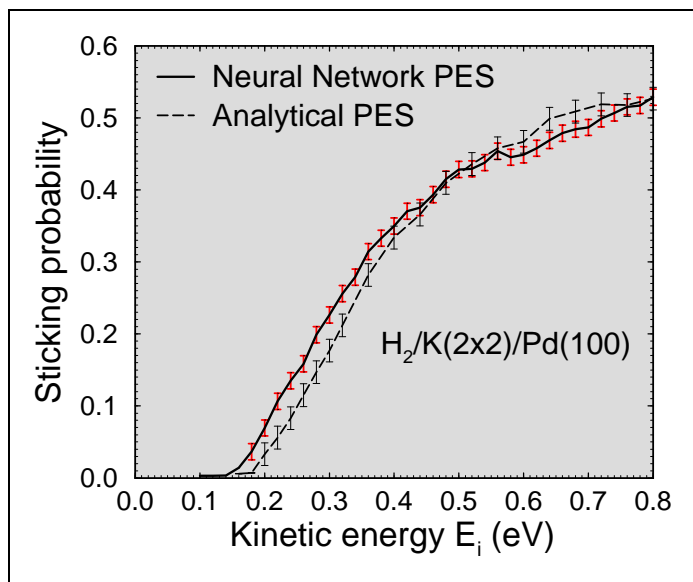
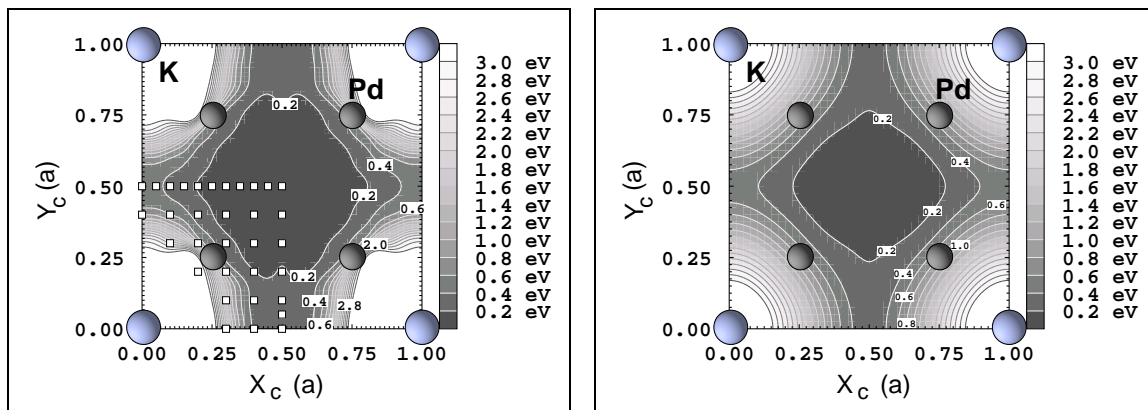


Figure 10.14: Sticking probability versus kinetic energy of the dissociation of  $H_2$  on a  $K(2 \times 2)/Pd(100)$  surface. The classical molecular dynamics calculations have been performed on a neural network and an analytical interpolation of the *ab initio* calculated energies. The error bars of the probabilities are also indicated. The neural network sticking curve is based on a total number of 76000 trajectories.

A closer look at the probabilities reveals that around 0.3 eV kinetic energy the difference between the two sticking curves is slightly larger than the statistical errors. This might be influenced by the form of the corrugation. The analytical interpolation is based on the *ab initio* data of the elbow plots only. The corrugation has not been used for the minimisation of the parameters in the analytical fit. The neural network corrugation and the analytical corrugation are presented in Fig. 10.15, cf. the *ab initio* cut in Fig. 10.9, p. 122. Considering the cut through the neural network PES and also



(a) Neural Network corrugation

(b) Analytical corrugation

Figure 10.15: Corrugation of the energy across the unit cell. The molecule is oriented parallel to the surface with its bond axis parallel to  $Y_c$  at a height  $z=1.9 \text{ \AA}$  with a bond length  $d=0.77 \text{ \AA}$  as plotted in Fig 10.8, p. 10.8. (a) and (b): potential energies as a function of the lateral coordinates ( $X_c$ ,  $Y_c$ ). The contour spacing is 0.2 eV. The white squares in (a) mark the calculated *ab initio* points.

through the *ab initio* PES, the area of the unit cell with a barrier lower than 0.3 eV is 9% percent larger compared to the analytical cut. Furthermore, it is visible that the analytical corrugation - being not fitted to the *ab initio* data - increases slightly slower than the NN-PES moving from a Pd on-top position to a K on-top site. This differences may influence the adsorption process. The more open structure of the *ab initio* and neural network cut may lead to the slightly higher sticking probability at low kinetic energies. On the other hand, the less pronounced corrugation of the analytical cut could cause the slightly higher sticking coefficient at high energies. However, the differences in the sticking probabilities are too small to draw a conclusion from one two-dimensional cut through the six-dimensional configuration space. The agreement between the two interpolation methods for this system is in fact very good.

In Tab. 10.3 we compare some sticking probabilities of the general neural network with the constrained neural network fitting scheme. The two different approaches also agree well, the differences do not exceed the statistical errors.

### 10.5.1 $H_2$ dissociation on K/Pd(100) versus S/Pd(100)

We are now able to compare the dynamical consequences of the adsorption of potassium and sulphur atoms concerning the dissociation of hydrogen on the Pd(100) surface. In Fig. 10.16 we plotted the sticking probabilities for both systems as a function of the kinetic energy. In addition, also the integrated barrier distribution  $P_b(E)$ ,

$$P_b(E) = \frac{1}{2\pi A} \int \Theta(E - E_b(\theta, \phi, X, Y)) \cos \theta \, d\theta \, d\phi \, dX \, dY \quad (10.2)$$

Kinetic energy	Sticking probability based on the	
	General NN-PES	Constrained NN-PES
0.2 eV	6.9 %	3.6 %
0.3 eV	22.7 %	24.8 %
0.5 eV	42.9 %	44.8 %
0.8 eV	52.9 %	52.7 %

Table 10.3: Comparison of the sticking probabilities of the general and the constrained neural network interpolation approach for  $H_2/K/Pd(100)$  under normal incidence.

is shown. Here  $A$  is the area of the surface unit cell and  $E_b$  is the minimum energy barrier along a cut defined by the quadruple  $(\theta, \phi, X, Y)$ . The quantity  $P_b(E)$  is the fraction of the configuration space for which the barrier towards dissociation is less than  $E$ . It corresponds to the sticking probability in the classical sudden approximation or the so-called “hole model” [144].

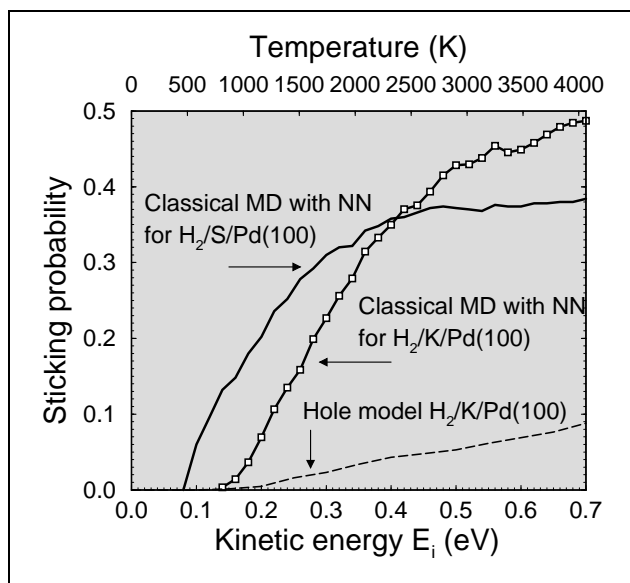


Figure 10.16: Sticking probability versus kinetic energy for hydrogen dissociation on a K and S covered Pd(100) surface under normal incidence. Also the so-called hole model or integrated barrier distribution has been plotted.

The onset of dissociative adsorption is lower for the sulphur covered sample, which is in agreement with the lower minimum energy barrier. Hence, at low kinetic energies or temperatures of the  $H_2$  beam for up to 2000 K, the Pd surface poisoned by sulphur atoms is more reactive than the same surface with potassium co-adsorbates. At around 0.4 eV and an adsorption probability of 35 %, the two sticking curves cross. For higher energies or temperatures the K-covered surface is more reactive. At a kinetic energy of 0.7 eV the system K/Pd(100) dissociates 50 % of the approaching hydrogen. With sulphur co-adsorbates the probability is 10 % lower. We point out that “reactive” relates in this context to the dissociation of a diatomic molecule. A catalytic reaction consists of a concert of events for which the dissociation process is indeed often the rate limiting step. However, after the dissociation the adsorbed particles should usually be available for the next step of the process, e.g. the reaction with another ad-particle on the surface.

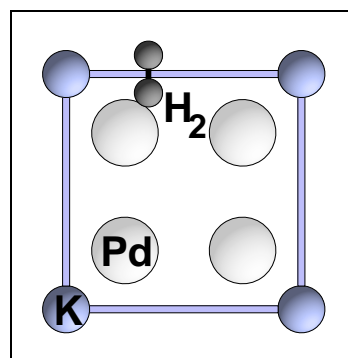


In this respect we note that due to the presence of K-adsorbates the adsorbed H-atoms are more strongly bound to the surface. Potassium co-adsorbates do not only poison the dissociation process of  $\text{H}_2$ , they also hinder the next step of the reaction.

The crossing of the two sticking curves can be understood in the context of the “steering effect”. Although the dissociation of hydrogen on the potassium and the sulphur covered Pd(100) surface in contrast to the clean surface is hindered by energy barriers, there is also significant steering of the impinging molecules operative. Molecules approaching the surface in a geometry in which they would not be able to dissociate can be re-oriented and re-directed by the forces to low barrier sites where they adsorb dissociatively. For the sulphur covered sample this has been shown already [51,62]. For the system K/Pd(100) the steering effect is reflected by the fact that the calculated sticking probabilities are much larger than what one would expect from the hole model [144] (see Fig. 10.16). In our discussion of the influence of the corrugation on reaction probabilities like the sticking coefficient in Sec. 9.5 we demonstrated that a stronger corrugation leads to a suppression of the steering effect. If the landscape on which the molecule moves is very curvy, the re-orientation of the particle is less effective; especially at larger kinetic energies the molecule hits a repulsive wall before it is able to reach a more favourable configuration. Now, the *ab initio* calculations revealed that the corrugation after co-adsorption of potassium on the Pd(100) surface is much weaker than after adsorption of sulphur on the same surface. The road map is less curvy and hence steering can be operative at even higher energies. Consequently, the weaker corrugation leads to higher sticking probabilities in the high energy range.

### 10.5.2 Dissociation and scattering of $\text{H}_2$ on $\text{K}(2 \times 2)/\text{Pd}(100)$

With classical molecular dynamics simulations we are not only able to calculate an observable like the sticking probability, moreover they allow us to follow the motion of a molecule on a PES in detail. An understanding of the various processes connected to catalysis on an atomistic scale is essential for an improvement in the design of new catalysts. In Fig. 10.18 we have illustrated the motion of hydrogen molecules on the surface to more favourable adsorption geometries. The plot is divided into two parts: the left half relates to a dissociation event at an initial kinetic energy of the  $\text{H}_2$  molecule of 0.38 eV. The right half represents a scattering process at 0.6 eV. The initial configuration of the molecule at both energies is the same and is shown in Fig. 10.17. For simplicity, we started the MD run with the molecule oriented parallel to the surface with the lateral centre of mass coordinates  $(X_c, Y_c) = (0.35a, 1.0a)$ , where  $a$  is the lattice constant of the  $(2 \times 2)$  cell. Due to symmetry constraints, the molecule is not able to rotate in this configuration and the centre of mass can only move along the K-K line. We focus on the re-direction of the molecule. The re-orientation has been already discussed in the first



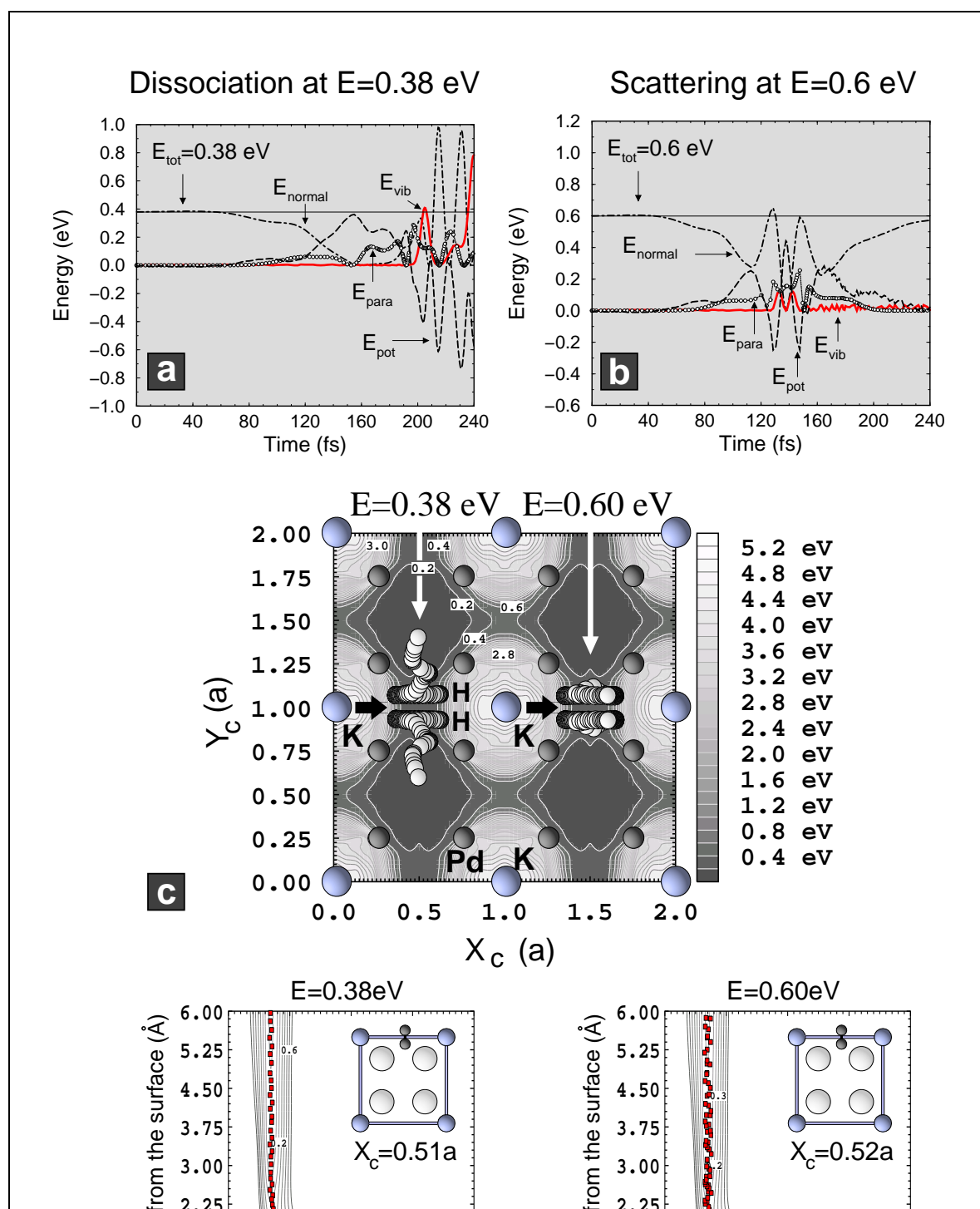
**Figure 10.17:** Initial configuration of the hydrogen molecule for the classical molecular dynamics runs plotted in Fig. 10.18.

chapter of this work. In its initial geometry the molecule is not able to dissociate for both kinetic energies. The PES is repulsive since the H-atoms point at the Pd surface atoms and the centre of mass is close to the K adsorbate (see Fig. 10.17).

In the central graph of Fig. 10.18 we projected the motion of the hydrogen molecule onto the PES of a lateral cut through the configuration space corresponding to the barrier region ( $Z=1.9 \text{ \AA}$ ,  $d=0.77 \text{ \AA}$ ). For dissociation, the molecule needs to traverse one point of this cut at a certain set of lateral centre of mass coordinates. The black arrows in Fig. 10.18(c) mark the initial position of the molecule far away from the surface. The white circles indicate the position of the two H-atoms with an initial energy of 0.38 eV (left) and 0.6 eV (right) at different stages of the process. The barrier corresponding to the initial configuration is too high for both energies to allow dissociation.

We will focus first on the molecule with the lower initial kinetic energy. The total  $\text{H}_2$  energy and its different contributions as a function of the time of the event are shown in Fig. 10.18(a). Initially the total energy is equal to the normal energy of the molecule corresponding to normal incidence. The rotational energy is always zero due to symmetry constraints. During the first 80 fs the molecule approaches the surface from the gas phase but does not change its lateral position significantly. After this period the potential energy rises and some of the normal energy is transferred into parallel energy of the molecule. It moves down the hill along the K-K line in (c). In the middle of the trench at  $(X_c, Y_c) = (0.5 a, 1.0 a)$  the molecule would have enough energy to overcome the corresponding barrier, but it is too fast to transfer energy into the vibrational mode and to stretch the bond sufficiently. The centre of mass continues to move further along the K-K line. After around 160 fs it has nearly reached the opposite site to its initial configuration and the parallel energy in (a) becomes zero [ $(X_c, Y_c) = (0.61 a, 1.0 a)$ ]. At this point the bond length is already stretched from 0.75  $\text{\AA}$  to 0.79  $\text{\AA}$ . The motion then changes its direction and the molecule approaches again the middle of the trench. After 190 fs, with the bond now sufficiently stretched, it re-gained enough kinetic energy to overcome the barrier. The potential energy starts to drop. At the same time the vibrational energy rises and the hydrogen bond starts to break. During the following time the molecule moves back and forth with respect to the surface (see below) which is indicated by the strongly varying potential energy. After 220 fs its value stays negative only and the vibrational energy rises steadily. At around 240 fs  $\text{H}_2$  is dissociated, the bond length is stretched by more than 400%. The two H-atoms have moved close to the centre of the  $(2 \times 2)$  cells as plotted in (c). In graph (d) we projected the motion of the centre of mass coordinate onto the elbow plot, i.e. onto the  $Zd$ -cut, in which the molecule overcomes the barrier in the entrance channel.

Figure 10.18: Dissociation and scattering of a  $H_2$  molecule on a  $K(2 \times 2)/Pd(100)$  surface at an initial normal energy of 0.38 eV (left part) and 0.6 eV (right part), respectively. The  $H_2$  molecule is both times initially oriented parallel to the surface with its bond axis along the  $Y_c$  direction at  $(X_c, Y_c) = (0.35 a, 1.0 a)$  as indicated by the black arrow in (c). In Fig. (a) and (b) the different energy contributions as a function of the time of the molecular dynamics run is shown. Fig. (c) displays the corrugation of the PES at a distance  $Z_c = 1.9 \text{ \AA}$  and a bond length of  $d = 0.77 \text{ \AA}$ . The motion of the two H-atoms has been projected into the contour plot of Fig. (c). Fig (d) and (e) show the variation of the centre of mass of the molecule projected onto the elbow plot in which (d) the molecule dissociates or (e) is scattered back into the gas phase.



The corresponding lateral position is  $(X_c, Y_c) = (0.51 a, 1.0 a)$ , i.e. close to the middle of the trench between the K-atoms. The minimum barrier in this configuration is located at a distance of  $1.8 \text{ \AA}$  to the topmost palladium layer and has a value of  $0.26 \text{ eV}$ . After having traversed the bottle-neck, the normal and parallel energy is transferred into the vibrational mode. The centre of mass is directed around the curve of the elbow plot and the molecular bond stretches further. We see in the lower right part of Fig. 10.18(d) that the centre of mass oscillates around the valley of the elbow plot. This situation corresponds to the oscillations of the potential energy in Fig. 10.18(a) starting after 200 fs and is a consequence of the curvature of the PES in this cut together with the kinetic energy of the molecule. The H-H distance continues to increase and  $\text{H}_2$  dissociates leaving its fragments on the surface.

At a higher kinetic energy of  $E = 0.6 \text{ eV}$  the re-direction of the molecule to more favourable dissociation geometries as shown in the previous example is also present. Again, for a certain time, here 50 fs, the molecule approaches the surface and does not change its lateral position. After this interval, when the potential energy starts to rise as shown in Fig. 10.18(b), the centre of mass begins to move from one side of the trench between two potassium atoms to the other side. However, the major difference is that the molecule is now too fast for a sufficient transfer of the parallel energy into the internal degrees of freedom. At the lateral coordinates  $(X_c, Y_c) = (0.52 a, 1.0 a)$  the molecule is again able to overcome the barrier in the entrance channel. But as shown in the corresponding elbow plot in Fig. 10.18(e), it hits a repulsive wall of the PES *before* it was traversed around the curve of the elbow. This takes place after 130 fs. The hydrogen molecule is then scattered back into the gas phase, but slightly vibrationally excited reflected by the oscillations of the centre of mass in Fig. 10.18(e). 240 fs after the molecule started to approach the surface the major contribution to the total energy is again the normal energy, but now the molecule moves away from the surface with its bond still intact.

The two trajectories reveal that steering is indeed operative on the  $\text{K}(2 \times 2)/\text{Pd}(100)$  surface. But although it is present even at high kinetic energies, it becomes less effective for fast molecules. It is the interplay of the different degrees of freedom of the impinging particle with the topology of the surface which determines the reaction probability. It is obvious that a static theory is not able to describe the reactivity of such processes on a catalytic surface. This demonstrates the importance of a reliable, high-dimensional potential-energy surface. The neural network interpolation of *ab initio* energies combined with molecular dynamics provides us with a tool that allows us to gain insight into dynamical processes on an atomistic scale. In addition, since the neural network output function is just an analytical function and fast to evaluate, the neural network is ideally suited for problems where extensive statistics are required.

---

## Chapter 11

### Neural Network application: Photodesorption of CO from $\text{Cr}_2\text{O}_3$

---

Dissociative adsorption, the bond breaking of a molecule upon adsorbing on a surface is one of the crucial steps in heterogenous catalysis. Another important part of a catalytic reaction is the desorption of a molecule from a surface. As a further application in addition to dissociation processes on metal surfaces we will now present the neural network interpolation of *ab initio* data for the photodesorption of CO from  $\text{Cr}_2\text{O}_3(0001)$ . CO adsorbs molecularly on the  $\text{Cr}_2\text{O}_3(0001)$  surface. State resolved laser induced desorption of small molecules from surfaces has been the subject of numerous experimental studies in the last years. The process is illustrated in Fig. 11.1. An adsorbed molecule on a surface is excited by a laser pulse and then desorbs into the gas phase. For the system CO/ $\text{Cr}_2\text{O}_3$  it has been found experimentally that CO molecules with a low rotational excitation desorb like a helicopter, i.e. with the rotational axis perpendicular to the surface, whereas high rotational excitations lead to a cartwheel like desorption [145].

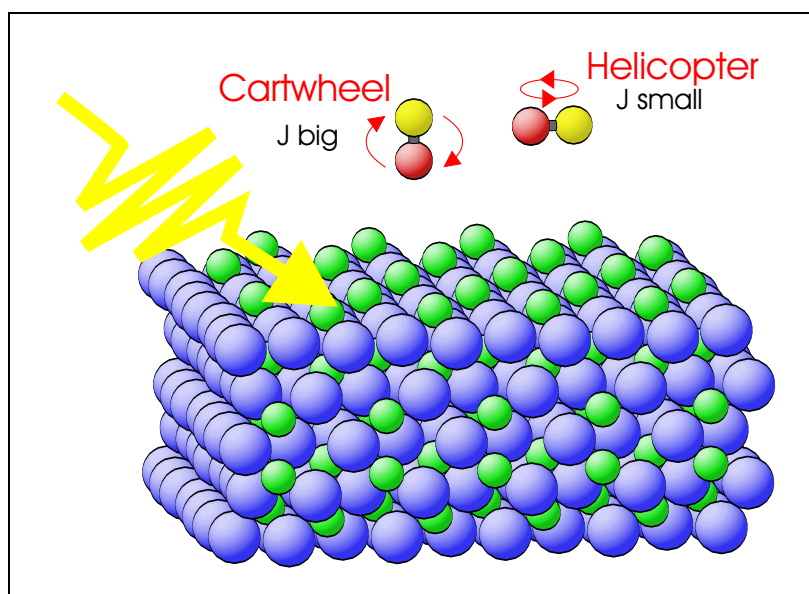


Figure 11.1: Laser induced desorption of CO from  $\text{Cr}_2\text{O}_3(0001)$ . The CO molecules desorb cartwheel- or helicopter-like, depending on their angular momentum.

Only recently it has been possible to gain fundamental theoretical insight into the mechanisms and driving forces of such DIET (Desorption Induced by Electronic Transition)-processes on oxide surfaces based on *ab initio* calculations for the first time [146]. For a theoretical study of such processes it is necessary to determine the ground state potential-energy surface of the molecule/surface system and also an excited state PES corresponding to the laser induced transition. The process is simulated dynamically by transferring a wave packet after propagating it for a certain residence lifetime on the excited state potential onto the electronic ground state PES. The wave packet is then propagated under the influence of the ground state PES and the desorbing part of the wave packet is analysed. This has been done successfully for the desorption of NO from the NiO(100) surface by Klüner and Freund [146]. Such a dynamical study requires the interpolation of the *ab initio* calculated PESs. For the system NO/NiO(100) analytical functions have been employed for this purpose.

The results presented in this chapter for the desorption of CO from Cr<sub>2</sub>O<sub>3</sub>(0001) have been determined in a collaboration with the Department of Chemical Physics (CP) at the Fritz-Haber-Institute in Berlin. The *ab initio* calculations and the wave packet molecular dynamics have been performed by Pykavy and Thiel. The details of the theoretical investigation can be found elsewhere [147, 148, 149]. We provided the interpolation of the *ab initio* ground state potential-energy surface with neural networks in order to make a dynamical study feasible. Independently, an analytical fit of the same *ab initio* PES has been evaluated [148]. This allows us to compare the reliability of the neural network potential within dynamical calculations not only to experimental data but also to results based on the analytical PES.

We point out that a theoretical study of the photodesorption process in addition to the ground state PES requires the evaluation of an excited state PES. The following dynamical results are preliminarily in that respect that a simple analytical dependency has been assumed for the latter. However, in both the calculation with the analytical and the neural network PES the same excited state potential has been used. Therefore, we are able to compare both approximations. Recently, also the excited state has been calculated with *ab initio* methods [150] and it is planned to extend the neural network approach to the corresponding PES as well.

## 11.1 Laser induced desorption experiment

CO adsorbs molecularly, i.e. it does not dissociate on Cr<sub>2</sub>O<sub>3</sub>(0001). In the laser induced desorption experiment pulses of 6.4 eV with a wavelength of  $\lambda = 193$  nm have been used for 15 ns to excite and desorb the molecules as illustrated in Fig. 11.1. The desorbing molecules have then been detected quantum state resolved [145].

The experimental study focused on the rotational alignment of the desorbing molecules, i.e. the direction of angular momentum as a function of the rotational quantum number. The principle of the experiment is to measure the absorption coefficient depending on different polarisations of the laserlight. The effects can be quantified by the determination of the quadrupolemoment of the desorbing CO molecule. A positive quadrupolemoment of  $A_0 = 2$  reflects a pure helicopter-like molecule, a value of  $A_0 = -1$

corresponds to a pure cartwheel-like rotation. The quadrupolemoment as a function of the rotational quantum number has been plotted in Fig. 11.2 for two different desorption velocities. We will concentrate on the results for the high desorption velocity  $v = 1160$  m/s. For rotational quantum numbers up to  $J < 25$  the quadrupolemoment has a value around  $A_0 = 0.5$ , whereas at higher numbers the quadrupolemoment drops until it reaches a value of  $A_0 = -1$  at  $J = 35$ . The interpretation of this behaviour is that slowly rotating molecules exhibit a desorption behaviour which is more like a helicopter motion. At higher  $J$  the angular momentum changes into a direction parallel to the surface plane. For the lower desorption velocity in Fig. 11.2 the shape of the curve is similar but shifted to lower rotational quantum numbers.

Hence, two different modes of desorption for the CO molecule exist: One corresponding to the helicopter, the other to the cartwheel-like motion. The measurement of such vector quantities of a desorbate is important since it enlarges the knowledge about the principles of desorption in addition to the usually measured and calculated scalar quantities like desorption probabilities.

Furthermore, the velocity distributions of the desorbing molecule in the vibrational ground state have been measured. We will concentrate on the width of the distributions in Fig. 11.3. For all angular momentum quantum numbers the velocity ranges from values of 0 to 2000 m/s.

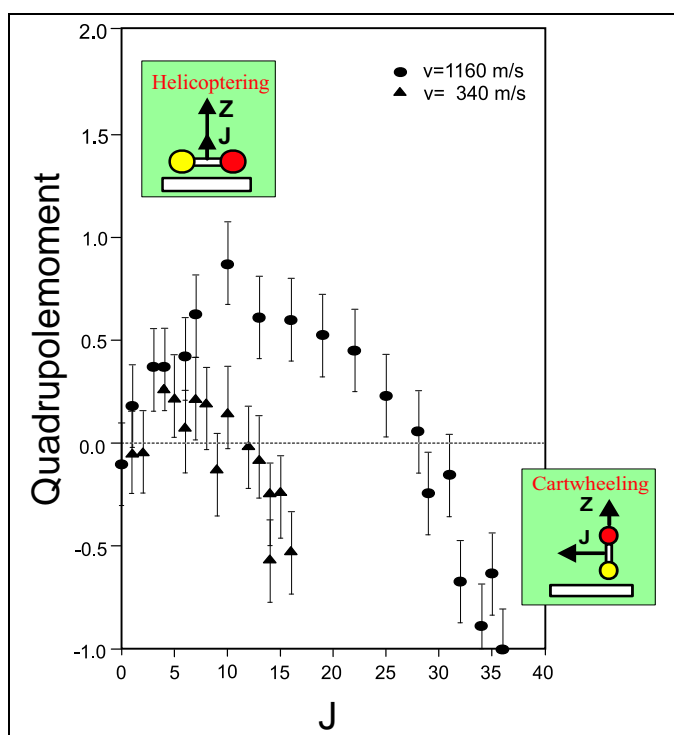


Figure 11.2: *Experiment: Quadrupolemoment versus rotational quantum number  $J$  for the system CO/Cr<sub>2</sub>O<sub>3</sub>. At high angular quantum number CO desorbes like a cartwheel, at low  $J$  the motion is more helicopter-like.*

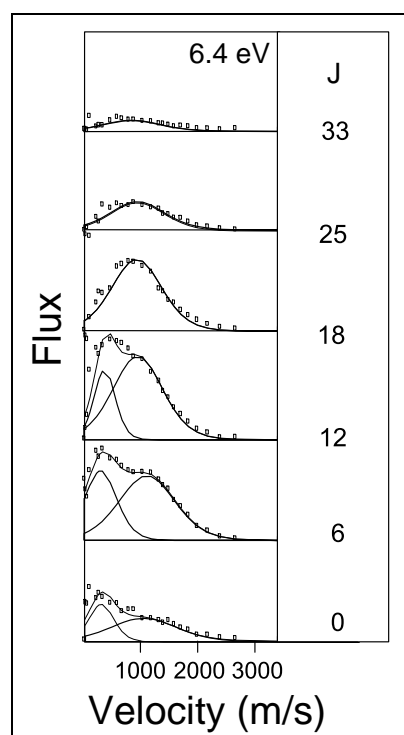


Figure 11.3: *Experiment: Velocity distribution as a function of the rotational quantum number  $J$ . The molecule is in the vibrational ground state.*

## 11.2 *Ab initio* MD with Neural Networks

The *ab initio* ground state PES of the CO molecule on Cr<sub>2</sub>O<sub>3</sub> has been determined with a Hartree-Fock cluster calculation [147, 148, 149]. The PES is three-dimensional depending on the distance of the centre of mass from the surface  $Z$  and the angles  $\theta$  and  $\phi$ . In order to include the symmetry of the problem we transformed the three coordinates into the following inputs to the neural network:

- $x_1 = e^{-Z}$ ,
- $x_2 = e^{-Z} \cos(\phi) \sin(\theta)$ ,
- $x_3 = e^{-Z} \theta$ ,

where  $Z$  ranges between 2.0 and 15.0  $a_0$  ( $a_0 = 0.529 \text{ \AA}$ ),  $\phi$  is calculated between 0–360 degrees and  $\theta$  varies from 0–180 degrees. Again, the term  $e^{-Z}$  has been used to reflect that the molecule in the vacuum far away from the surface should only depend on the bond length of the molecule and the influence of all other coordinates should vanish. The molecule is non-vibrating, i.e. the bond length is not included in the interpolation. Furthermore, since the diatomic molecule consists of different species an azimuthal symmetry of  $\cos(\phi)$  has been included instead of the  $\cos(2\phi)$  term for the hydrogen molecule. This term is weighted by the factor  $\sin(\theta)$  since an upright molecule should not exhibit any azimuthal dependency. We did not apply any  $\theta$  symmetry for the third neural network input  $x_3$ .

745 *ab initio* total energies of the ground state PES have been determined which are divided into a training set of 652 and a test set of 93 points. Different neural network fits have been performed which are listed in Tab. 11.1. We obtained the best result with a 3–40–40–1 *tl* network and achieved a test and training error of 0.017 eV and 0.03 eV, respectively. All the errors lie within the desired *ab initio* accuracy of 0.1 eV. The interpolation time on one IBM-SP2 node ranged from several minutes for the smallest network to several hours for the most complex one.

Neural Network	Number of weights	Root mean squared error (RMSE)	
		Training Set	Test Set
3–10–10–1 <i>tl</i>	161	46 meV	72 meV
3–20–20–1 <i>tl</i>	521	23 meV	52 meV
3–30–30–1 <i>tl</i>	1081	28 meV	54 meV
3–40–40–1 <i>tl</i>	1841	17 meV	30 meV
3–50–50–1 <i>tl</i>	2801	21 meV	44 meV

Table 11.1: Root mean squared error for different Neural Network interpolations of the ground state PES for the system CO/Cr<sub>2</sub>O<sub>3</sub>. The training set consists of 652 *ab initio* total energies, the test set of 93. NN-Parameter:  $\lambda(0) = 0.98$ ,  $\lambda_0 = 0.99926$ ,  $a_{th} = 0.6$ .



The CO molecule adsorbs on the  $\text{Cr}_2\text{O}_3$  surface preferentially with its axis nearly parallel to the surface. The favourable adsorption site is the Cr bridge site with the C and O atoms of the molecule oriented towards the Cr atoms. This has been drawn schematically in Fig. 11.4. The neural network energy as a function of the CO distance from the surface  $Z$  and the azimuthal angle  $\phi$  above the Cr bridge site has been plotted in Fig. 11.5. The energy minimum is located at a distance of  $4.5 a_0$  and an angle  $\phi = 180^\circ$ .

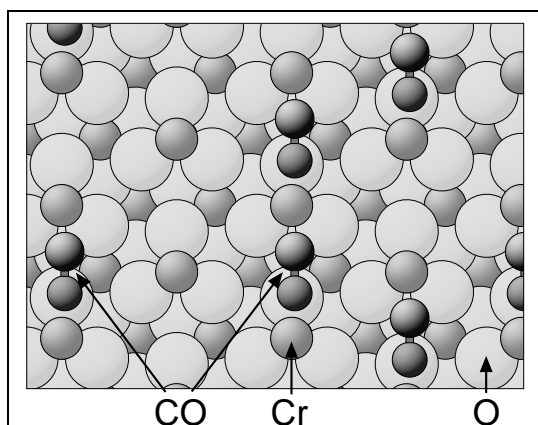


Figure 11.4: Top view of the  $\text{Cr}_2\text{O}_3$  surface with a number of adsorbed CO molecules. The most favourable adsorption site is the Cr bridge site with the molecule axis parallel to the surface and the C and O atoms oriented towards the Cr atoms.

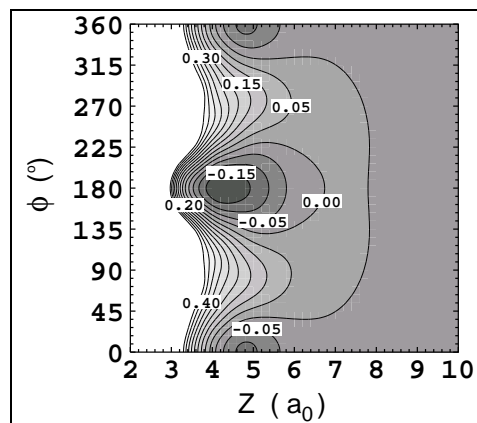


Figure 11.5: 2D cut through the 3D NN-PES of CO/ $\text{Cr}_2\text{O}_3$ . CO is adsorbed on the Cr bridge site as in Fig. 11.4. The polar angle is fixed at  $\theta = 120^\circ$ . The minimum is located at ( $Z = 4.5 a_0$ ,  $\phi = 180^\circ$ ). Energies: in eV; Contour spacing: 0.05 eV.

Besides the neural network interpolation of the ground state PES an analytical fit has been determined independently by the theory group of the CP-Department. Wave packet dynamics has been performed on the analytical and the neural network PES. A resonance lifetime of  $\tau = 70$  fs has been used for the movement on the excited state PES. Fig. 11.6 displays the average value of the quantum number  $M$  as a function of the rotational quantum number  $J$ . Here,  $M$  is the length of  $J_Z$  vector, i.e. the length of the projection of the rotational momentum on the  $Z$ -axis. For a helicopter-like molecule the direction of the angular momentum vector is perpendicular to the bond axis, i.e. along the  $Z$ -axis and we have  $M = J$ . This is illustrated in insets of Fig. 11.6 and the described situation corresponds to the dashed line in the same figure. For a cartwheel-like motion  $J$  is perpendicular to the  $Z$ -axis and  $M = 0$ .

For low rotational quantum numbers the values in Fig. 11.6 are indeed close to the dashed line of the pure helicopter motion. At high  $J$  they are far away from this line and decrease steadily reflecting a more cartwheel-like desorbing molecule. The region of an intermediated motion starts at a value of  $J > 30$  and therefore in the region where the same transition has been found in the experimental study, see Fig. 11.2, p. 139. There are small differences visible for the calculations based on the analytical and the neural network ground state PES, but their overall trend is similar. Both fits describe the desorption of CO in the asymptotic regions correctly. Furthermore,

the calculated velocity distribution in Fig. 11.7 for the non-vibrating molecule moving on the analytical and the neural PES agree also very well. As in the experiment the velocities range from 0–2000 m/s. The calculated velocity distributions are averaged over the rotational quantum numbers  $J$ .

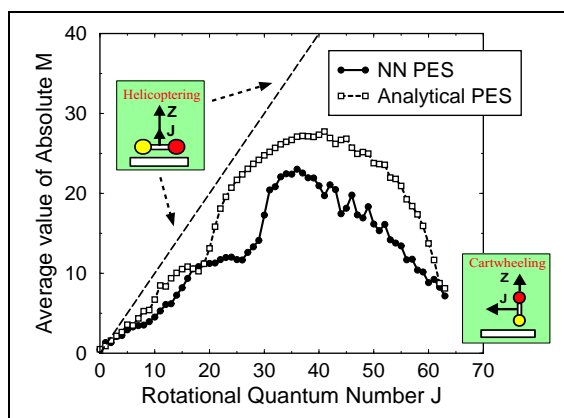


Figure 11.6: *Theory: Rotational alignment as described by the average value of  $M$  as a function of the rotational quantum number  $J$ .  $M$  is the length of the projection of the angular momentum vector onto the  $Z$ -axis.*

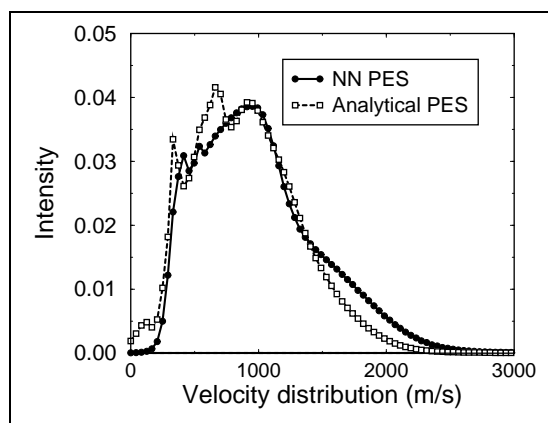


Figure 11.7: *Theory: Velocity distribution from wave packet dynamics based on a Neural Network and an analytically interpolated *ab initio* ground state PES. The velocities are averaged over the rotational quantum numbers  $J$ .*

The system will be subject to further investigation. The *ab initio* calculations have been extended to the excited state PES and moreover both potential energies have now been determined depending on one more degree of freedom, the  $X$  coordinate along the Cr-Cr axis.

To conclude, both the analytical and the neural network ground state PES give a qualitative agreement with the experiment and support the existence of two different desorption modes of the CO-molecule. The main advantage of the neural network interpolation over the analytical one is its universality. We used the same program which has been employed in dissociation reactions. We only needed to adjust the pre-processing of the input data reflecting the symmetry of the problem. The training of the neural network took only a few hours.

In the case of the analytical fit a higher insight was necessary to choose appropriate basis functions and the adaptation of an existing computer program to describe the discussed desorption system was required. This procedure takes at least several days depending on experience. Furthermore, the extension of the fit to higher dimensions, i.e. a 4D approximation, is no fundamental problem to the neural network approach. One simply has to use a network with four input nodes and needs to adjust the pre-processing and is ready to start. Yet, the analytical fitting procedure needs to be revised to include a new coordinate which requires further insight and will lack transferability to other systems not to mention physical problems.

---

## Chapter 12

### Summary

---

The objective of this work was to develop an alternative interpolation method for *ab initio* potential energy surfaces using feedforward neural networks and to apply the new method in classical molecular dynamics calculations of surface reaction rates. In particular we were interested in the poisoning effect of potassium adsorbates to the dissociative adsorption of hydrogen molecules on a Pd(100) metal surface. The work was motivated by the fact that due to the statistical nature of dissociation processes *ab initio* molecular dynamics do not allow the determination of reaction probabilities. For the calculation of the sticking probability in dissociative adsorption reactions we followed a three step approach:

1. the determination of the *ab initio* potential energy surface by density functional theory methods, i.e. the FP-LAPW method,
2. a fit of the calculated total energies to a continuous neural networks representation including all relevant degrees of freedom,
3. a molecular dynamics calculation on the neural network representation of the *ab initio* PES with extensive statistics.

We showed that neural networks can interpolate *ab initio* potential energy surfaces of several degrees of freedom accurately. The approximation error on the fitted as well as the non-fitted total energies does not exceed the accuracy of the underlying *ab initio* calculations. The computational cost of training a neural network is small and just a fraction of the costs of the DFT calculations. The resulting neural network output function, the potential energy, and its derivatives, the forces, are very efficient to evaluate and allow molecular dynamics calculations with extensive statistics. Besides being fast and accurate, neural networks are general, i.e. they allow applications to a range of problems without writing a new code. This has been demonstrated by employing the neural network approximation approach in six-dimensional dissociative adsorption processes on metal surfaces as well as three-dimensional photodesorption of a diatomic molecule from a metal-oxide surface.

Concerning the amount of training data required to obtain a reliable representation it is not sufficient to perform a neural network fit based on the usually calculated top, bridge and hollow sites only. Intermediated configurations need to be considered. An *equidistant* sampling results in a number of  $10^4 - 10^5$  total energies for an accurate

interpolation. The required number of training energies for dissociation processes can be further reduced by an *effective* sampling of the configurations. Model calculations on the system  $\text{H}_2/\text{S}(2 \times 2)/\text{Pd}(100)$  revealed, that the form of the energetic corrugation can influence the dynamical result greatly. We therefore proposed a modification of the usually applied sampling of total energies in DFT calculations of dissociation processes. In addition to elbow plots above high-symmetric sites we recommend to calculate the corrugation of the PES in more detail by collecting information of the potential energy as a function of the lateral coordinates within the surface unit cell. The modified sampling scheme allows to calculate dynamical results with neural networks based on  $10^3 - 10^4$  *ab initio* energies. The costs for a description of dissociation reactions with neural networks are orders of magnitude smaller compared to direct *ab initio* dynamics where up to  $10^7$  energies are necessary.

Following the initially proposed three step approach for *ab initio* molecular dynamics, neural networks have been applied to interpolate *ab initio* potential energy surfaces. The *ab initio* PES for the dissociation of hydrogen on a potassium covered Pd(100) surface has been studied in detail by density functional theory and the FP-LAPW method. It was found that potassium adsorbates on Pd(100) poison the dissociation of hydrogen molecules by the formation of energy barriers in the entrance channel, but in the exit channel they tend to promote adsorption of H-atoms on the surface. These findings were in agreement with experiments. The corrugation in the bond-breaking region was found to be much less pronounced on the potassium covered surface than on the sulphur covered sample except for the vicinity of the K or S atoms.

Classical molecular dynamics simulations based on a six-dimensional neural network representation of 659 *ab initio* energies provided further support to the importance of dynamical steering. Overall 76000 trajectories have been calculated. The combined DFT and neural network approach for this system allowed us to present a theoretical comparison of the dynamical consequences of the adsorption of an electropositive adsorbate like potassium and an electronegative like sulphur on the dissociation of  $\text{H}_2$  on the same substrate. At initial kinetic energies lower than 0.4 eV the sulphur covered Pd(100) surface was found to be more reactive concerning hydrogen dissociation than the potassium covered sample. At higher kinetic energies or temperatures of the hydrogen beam the situation reverses. We were able to associate these findings with the form of the corrugation of the potential energy for both surfaces. In addition, a detailed investigation and analysis of single trajectories on the  $\text{H}_2/\text{K}(2 \times 2)/\text{Pd}(100)$  neural network PES on an atomistic scale underlined the importance of high-dimensional dynamical studies. It is the interplay of the different degrees of freedom of the impinging particle with the topology of the surface which determines the reaction probability.

As a further application in addition to dissociation processes on metal surfaces we studied the neural network interpolation of *ab initio* data for the photodesorption of CO from  $\text{Cr}_2\text{O}_3(0001)$ . This work has been performed in collaboration with the Department of Chemical Physics at the Fritz-Haber-Institute in Berlin. The rotational alignment of the desorbing molecule has been studied by wave-packet dynamics based on the three-dimensional neural network PES and also an analytical PES. The dynamical results on both representations showed that slowly rotating molecules exhibit a desorption behaviour which is more like a helicopter motion, whereas fast rotating

molecule desorb like a cart-wheel. These findings were in agreement with laser-induced desorption experiments.

To conclude, the neural network interpolation of *ab initio* energies combined with molecular dynamics provides us with a tool to study reaction processes where extensive statistics are required. In addition, neural networks preserve the accuracy of DFT calculations and allow us to gain insight into dynamical processes on an atomistic scale. In recent years, caused by the development of efficient algorithms and the increase of computer power, it became possible to obtain detailed potential energy surfaces by density functional theory calculations. *Ab initio* based calculations have enormously enlarged our knowledge about the properties of materials and processes on surfaces. Still, quantitative simulations of reactions on surfaces are still limited to rather simple systems. Six- or seven dimensional classical and quantum dynamics based on DFT are the cutting-edge of dynamical studies. Yet, the structure of neural networks is not limited with respect to the input degrees of freedom. When detailed higher-dimensional potential energy surfaces based on DFT calculations become available, it will be interesting to extend the neural network studies to problems involving more than six degrees of freedom in the future.



**Part IV.**  
**Appendix**





---

## Chapter A

### Derivation of Backpropagation

---

For the update of the weights in gradient-descent based algorithms the partial derivatives of the error function with respect to the weights need to be calculated. The cost function of the neural network is the sum over the squared residuals of the target value  $t_k$  and the current value  $y_k^{N=2}$  of the  $k$ -th output node for each example  $p$ :

$$E = \sum_p E_p(\mathbf{w}, \mathbf{y}^0) \quad \text{with:} \quad E_p = \frac{1}{2} \left( \sum_k (t_k - y_k^N(\mathbf{w}, \mathbf{y}^0))^2 \right). \quad (\text{A.1})$$

This cost function needs to be minimised with respect to the weights, i.e.:

$$\frac{\partial E}{\partial \mathbf{w}} = \sum_p \frac{\partial E_p(\mathbf{w}, \mathbf{y}^0)}{\partial \mathbf{w}} = 0. \quad (\text{A.2})$$

Therefore the computation of the derivatives  $\frac{\partial E_p}{\partial \mathbf{w}}$  is required. In order to derive this derivative, let us assume a three-layer network as plotted in Fig. 5.2, p. 44. In the network, the following relationships hold for the first and the second layer respectively, in which the input layer is counted as the 0-th layer:

$$\text{Input - hidden layer} \quad \begin{cases} x_j^1(\mathbf{w}, \mathbf{y}^0) = \sum_i^{n_0} w_{ij}^1 y_i^0 \\ y_j^1(\mathbf{w}, \mathbf{y}^0) = f_j^1(x_j^1) \end{cases}, \quad (\text{A.3})$$

$$\text{Hidden - output layer} \quad \begin{cases} x_k^2(\mathbf{w}, \mathbf{y}^0) = \sum_j^{n_1} w_{jk}^2 y_j^1 \\ y_k^2(\mathbf{w}, \mathbf{y}^0) = f_k^2(x_k^2) \end{cases}, \quad (\text{A.4})$$

where we dropped the bias terms for simplicity. The output of each neuron depends on the network input vector  $\mathbf{y}^0$  and the current weight vector  $\mathbf{w}$ . The partial derivative of the error function (A.1) with respect to a weight leading into an output node can be calculated using the chain rule:

$$\frac{\partial E_p}{\partial w_{jk}^2} = - (t_k - y_k^2(\mathbf{w}, \mathbf{y}^0)) \frac{\partial y_k^2}{\partial w_{jk}^2} = - (t_k - y_k^2(\mathbf{w}, \mathbf{y}^0)) \frac{\partial f^2}{\partial x_k^2} \frac{\partial x_k^2}{\partial w_{jk}^2}. \quad (\text{A.5})$$

Let us define the following:

$$\delta_k = \left( t_k - y_k^2(\mathbf{w}, \mathbf{y}^0) \right) \frac{\partial f^2}{\partial x_k^2} . \quad (\text{A.6})$$

With this and

$$\frac{\partial x_k^2}{\partial w_{jk}^2} = y_j^1 , \quad (\text{A.7})$$

we can rewrite equation (A.5) as:

$$\frac{\partial E_p}{\partial w_{jk}^2} = -\delta_k y_j^1 . \quad (\text{A.8})$$

Consequently, the partial derivative of the error function with respect to a weight leading into a hidden node is (with  $n_2$  output nodes):

$$\frac{\partial E_p}{\partial w_{ij}^1} = -\sum_{k=1}^{n_2} \left( t_k - y_k^2(\mathbf{w}, \mathbf{y}^0) \right) \frac{\partial f^2}{\partial x_k^2} \frac{\partial x_k^2}{\partial y_j^1} \frac{\partial f^1}{\partial x_j^1} \frac{\partial x_j^1}{\partial w_{ij}^1} . \quad (\text{A.9})$$

$$= -\sum_{k=1}^{n_2} \left( t_k - y_k^2(\mathbf{w}, \mathbf{y}^0) \right) \frac{\partial f^2}{\partial x_k^2} w_{jk}^2 \frac{\partial f^1}{\partial x_j^1} y_i^0 \quad (\text{A.10})$$

$$= -\frac{\partial f^1}{\partial x_j^1} y_i^0 \sum_{k=1}^{n_2} \delta_k w_{jk}^2 . \quad (\text{A.11})$$

If we further define:

$$\delta_j = \frac{\partial f^1}{\partial x_j^1} \sum_{k=1}^{n_2} \delta_k w_{jk}^2 , \quad (\text{A.12})$$

we get for  $\partial E_p / \partial w_{ij}^1$ , compare to equation (A.8):

$$\frac{\partial E_p}{\partial w_{ij}^1} = -\delta_j y_i^0 . \quad (\text{A.13})$$

These derivations can be generalised to networks with more than three layers. The result is the following set of formulas. The first one specifies the weight changes for weights leading into unit  $j$  in layer  $m$ , no matter what layer  $m$  actually is. The second formula specifies the error signal for the output layer  $N$  and the third formula gives the error signal for unit  $j$ , in a hidden layer with units  $k$  above:

$$\Delta w_{ij}^m = \eta \delta_j y_i^{m-1} , \quad (\text{A.14})$$

$$\delta_j^m = \left( t_k - y_k^m \right) \frac{\partial f^m}{\partial x_j^m} \quad m = N \quad (\text{output layer}) , \quad (\text{A.15})$$

$$\delta_j^m = \frac{\partial f^m}{\partial x_j^m} \sum_{k=1}^{n_{m+1}} \delta_k^{m+1} w_{jk}^{m+1} \quad m \neq N \quad (\text{hidden layer}) . \quad (\text{A.16})$$

---

## Chapter B

### Extended Kalman filter equations

---

We will follow here the derivation of the Kalman filter equations presented by Shah et al. [126]. However, we will apply different cost functions as proposed in the original paper. We will first introduce the Kalman filter equations with a weighting parameter for each example. In the second section we will then derive the equations for optimising the potential and the forces simultaneously.

#### B.1 Kalman filter equations with weighting

Let  $(\mathbf{x}(j), \mathbf{t}(j))$  be a sequence of nonrandom input/output pairs in iteration  $j$  of a network with  $T$  weights and  $L$  output nodes. We will consider *on-line* minimisation, i.e. we want to change the weights after presentation of each example. We view the synaptic weight vector  $\mathbf{w}$  as the state of a static nonlinear dynamic system described by the following equations:

$$\begin{aligned} \mathbf{w}(j+1) &= \mathbf{w}(j) = \mathbf{w}_{opt} \\ \mathbf{t}(j) &= \mathbf{y}(\mathbf{w}_{opt}, \mathbf{x}(j)) + \mathbf{e}(j), \end{aligned} \quad (\text{B.1})$$

where  $j$  is the time index,  $\mathbf{y}(\mathbf{w}_{opt}, \mathbf{x}(j))$  is the time-varying neural network output function and  $\mathbf{e}(j)$  is the sequence of modelling errors. The extended Kalman filter equations are derived by expanding the non-linear function  $\mathbf{y}(\mathbf{w}, \mathbf{x}(n))$  around the current estimate parameter vector  $\hat{\mathbf{w}}(n-1)$ , estimated from all data up to time  $(n-1)$ . The state model is accordingly:

$$\begin{aligned} \mathbf{w}(n) &= \mathbf{w}(n-1) = \mathbf{w}_{opt} \\ \mathbf{t}(n) &= \mathbf{y}(\hat{\mathbf{w}}(n-1), \mathbf{x}(n)) + \mathbf{J}^T(n) (\mathbf{w}_{opt} - \hat{\mathbf{w}}(n-1)) + \rho(n) + \mathbf{e}(n), \end{aligned} \quad (\text{B.2})$$

where  $\mathbf{J}(n)$  is the  $T \times L$  Jacobi matrix given by:

$$\mathbf{J}(n) = \left. \frac{\partial \mathbf{y}(\mathbf{w}, \mathbf{x}(n))}{\partial \mathbf{w}} \right|_{\mathbf{w}=\hat{\mathbf{w}}(n-1)}, \quad (\text{B.3})$$

and  $\rho(n)$  is the residual of the Taylor expansion of the network output function  $\mathbf{y}$ . The state model can be rewritten as:

$$\begin{aligned} \mathbf{w}(n) &= \mathbf{w}(n-1) = \mathbf{w}_{opt} \\ \mathbf{t}(n) &= \mathbf{J}^T(n) \mathbf{w}_{opt} + \xi(n) + \mathbf{e}(n), \end{aligned} \quad (\text{B.4})$$

with

$$\xi(n) = \mathbf{y}(\hat{\mathbf{w}}(n-1), \mathbf{x}(n)) - \mathbf{J}^T(n) \hat{\mathbf{w}}(n-1) + \rho(n) . \quad (\text{B.5})$$

The estimate  $\hat{\mathbf{w}}(n)$  is obtained by the optimal regression of  $\mathbf{w}_{opt}$ , i.e. we minimise:

$$\epsilon(n) = \sum_{j=1}^n \alpha(j) \|\mathbf{e}(j)\|_2 \lambda^{n-j} , \quad (\text{B.6})$$

where  $\alpha(j)$  is a weighting parameter for the examples in the data set. Minimisation means, we need to calculate:

$$\nabla_{\mathbf{w}_{opt}} \epsilon(n) = 2 \sum_{j=1}^n \alpha(j) \mathbf{J}(j) (\mathbf{t}(j) - \mathbf{J}^T(j) \mathbf{w}_{opt} - \xi(j)) \lambda^{n-j} = 0 . \quad (\text{B.7})$$

This gives after rearrangement:

$$\hat{\mathbf{w}}(n) = \Phi^{-1}(n) \mathbf{r}(n) , \quad (\text{B.8})$$

with:

$$\Phi(n) = \sum_{j=1}^n \alpha(j) \mathbf{J}(j) \mathbf{J}^T(j) \lambda^{n-j} , \quad (\text{B.9})$$

$$\mathbf{r}(n) = \sum_{j=1}^n \alpha(j) \lambda^{n-j} \mathbf{J}(j) (\mathbf{t}(j) - \xi(j)) . \quad (\text{B.10})$$

In order to get an iterative minimisation scheme we rewrite equation (B.9):

$$\Phi(n) = \lambda \sum_{j=1}^{n-1} \alpha(j) \lambda^{n-1-j} \mathbf{J}(j) \mathbf{J}^T(j) + \alpha(n) \mathbf{J}(n) \mathbf{J}^T(n) , \quad (\text{B.11})$$

$$\Phi(n) = \lambda \Phi(n-1) + \alpha(n) \mathbf{J}(n) \mathbf{J}^T(n) . \quad (\text{B.12})$$

Similarly we get from eq. (B.10):

$$\mathbf{r}(n) = \lambda \mathbf{r}(n-1) + \alpha(n) \mathbf{J}(n) (\mathbf{t}(n) - \xi(n)) . \quad (\text{B.13})$$

With the matrix inversion lemma:

$$\mathbf{A} = \mathbf{B}^{-1} + \alpha(n) \mathbf{C} \mathbf{C}^T , \quad (\text{B.14})$$

$$\mathbf{A}^{-1} = \mathbf{B} - \alpha(n) \mathbf{B} \mathbf{C} (\mathbf{I} + \mathbf{C}^T \mathbf{B} \mathbf{C})^{-1} \mathbf{C}^T \mathbf{B} , \quad (\text{B.15})$$

with  $\mathbf{B} = \Phi^{-1}(n-1) \lambda^{-1}$ , and  $\mathbf{C} = \mathbf{J}(n)$  we get:

$$\begin{aligned} \Phi^{-1}(n) &= \lambda^{-1} \Phi^{-1}(n-1) - \alpha(n) \lambda^{-2} \Phi^{-1}(n-1) \mathbf{J}(n) \\ &\quad \times \left[ \mathbf{I} + \alpha(n) \lambda^{-1} \mathbf{J}^T(n) \Phi^{-1}(n-1) \mathbf{J}(n) \right]^{-1} \mathbf{J}^T(n) \Phi^{-1}(n-1) . \end{aligned} \quad (\text{B.16})$$

We now define the matrix  $\mathbf{P}(n) = \Phi^{-1}(n)$ , and the Kalman gain matrix  $\mathbf{K}(n)$ :

$$\mathbf{K}(n) = \alpha(n) \lambda^{-1} \mathbf{P}(n-1) \mathbf{J}(n) \left[ \mathbf{I} + \alpha(n) \lambda^{-1} \mathbf{J}^T(n) \mathbf{P}(n-1) \mathbf{J}(n) \right]^{-1} . \quad (\text{B.17})$$

Equation (B.16) can be rewritten as:

$$\mathbf{P}(n) = \lambda^{-1}\mathbf{P}(n-1) - \lambda^{-1}\mathbf{K}(n)\mathbf{J}^T(n)\mathbf{P}(n-1) , \quad (\text{B.18})$$

and for equation (B.17) follows:

$$\mathbf{K}(n) \left[ \mathbf{I} + \alpha(n)\lambda^{-1}\mathbf{J}^T(n)\mathbf{P}(n-1)\mathbf{J}(n) \right] = \alpha(n)\lambda^{-1}\mathbf{P}(n-1)\mathbf{J}(n) , \quad (\text{B.19})$$

$$\mathbf{K}(n) + \alpha(n)\lambda^{-1}\mathbf{K}(n)\mathbf{J}^T(n)\mathbf{P}(n-1)\mathbf{J}(n) = \alpha(n)\lambda^{-1}\mathbf{P}(n-1)\mathbf{J}(n) , \quad (\text{B.20})$$

or

$$\mathbf{K}(n) = \left[ \alpha(n)\lambda^{-1}\mathbf{P}(n-1) - \alpha(n)\lambda^{-1}\mathbf{K}(n)\mathbf{J}^T(n)\mathbf{P}(n-1) \right] \mathbf{J}(n) . \quad (\text{B.21})$$

With equation (B.18) the Kalman gain matrix can now be expressed as:

$$\mathbf{K}(n) = \alpha(n) \mathbf{P}(n) \mathbf{J}(n) . \quad (\text{B.22})$$

The parameter estimate  $\hat{\mathbf{w}}(n)$  in (B.8) can now be written as:

$$\begin{aligned} \hat{\mathbf{w}}(n) &= \mathbf{P}(n) \mathbf{r}(n) , \\ &= \mathbf{P}(n)\lambda\mathbf{r}(n-1) + \alpha(n)\mathbf{P}(n)\mathbf{J}(n) (\mathbf{t}(n) - \xi(n)) . \end{aligned} \quad (\text{B.23})$$

Substituting (B.18) in the first term of (B.23) gives:

$$\begin{aligned} \hat{\mathbf{w}}(n) &= \mathbf{P}(n-1)\mathbf{r}(n-1) - \mathbf{K}(n)\mathbf{J}^T(n)\mathbf{P}(n-1)\mathbf{r}(n-1) \\ &\quad + \alpha(n)\mathbf{P}(n)\mathbf{J}(n) (\mathbf{t}(n) - \xi(n)) . \end{aligned} \quad (\text{B.24})$$

From (B.8) and (B.22):

$$\hat{\mathbf{w}}(n) = \hat{\mathbf{w}}(n-1) + \mathbf{K}(n) (\mathbf{t}(n) - \xi(n) - \mathbf{J}^T(n)\hat{\mathbf{w}}(n-1)) , \quad (\text{B.25})$$

$$\hat{\mathbf{w}}(n) = \hat{\mathbf{w}}(n-1) + \mathbf{K}(n) (\mathbf{t}(n) - \mathbf{y}(\hat{\mathbf{w}}(n-1), \mathbf{x}(n))) . \quad (\text{B.26})$$

The Kalman filter recursion for a network with  $T$  nodes and  $L$  output nodes are the equations (B.17), (B.18) and (B.26):

$$\mathbf{K}(n) = \alpha(n)\lambda^{-1}\mathbf{P}(n-1)\mathbf{J}(n) \left[ \mathbf{I} + \lambda^{-1}\mathbf{J}^T(n)\mathbf{P}(n-1)\mathbf{J}(n) \right]^{-1} , \quad (\text{B.27})$$

$$\hat{\mathbf{w}}(n) = \hat{\mathbf{w}}(n-1) + \mathbf{K}(n) (\mathbf{t}(n) - \mathbf{y}(\hat{\mathbf{w}}(n-1), \mathbf{x}(n))) , \quad (\text{B.28})$$

$$\mathbf{P}(n) = \lambda^{-1}\mathbf{P}(n-1) - \lambda^{-1}\mathbf{K}(n)\mathbf{J}^T(n)\mathbf{P}(n-1) . \quad (\text{B.29})$$

For minimisation we have to cycle through these equations until convergence of the weights in (B.28) is reached. The Kalman gain matrix  $\mathbf{K}$  is a  $T \times L$  matrix and the matrix  $\mathbf{P}$  is a  $T \times T$  matrix. The algorithm can be initialised with  $\mathbf{P}(0) = \delta^{-1}\mathbf{I}$ , with  $\delta > 0$  a small arbitrary value. Complexity is of the order  $LT^2$  multiplications per time step. One also has to compute the inverse of the  $L \times L$  matrix  $\left[ \mathbf{I} + \lambda^{-1}\mathbf{J}^T\mathbf{P}\mathbf{J} \right]$ . In our case with one output node  $L = 1$  this is just a division by a factor. The derivation of the Jacobi matrix in the neural network can be found in Appendix C, p. 157.

## B.2 Kalman filter equations with forces

Since the neural network output function is an analytical function of the inputs, the derivatives of the output with respect to the inputs can also be easily computed. In order to minimise the potential *and* the forces simultaneously, we rewrite the original error function in equation (B.6), p.152, as follows:

$$\epsilon(n) = \sum_{j=1}^n \alpha_j \left[ \|\mathbf{e}(j)\|_2 + \sum_{i=1}^N \left\| \frac{\partial \mathbf{t}(j)}{\partial x_i(j)} - \frac{\partial \mathbf{y}(\mathbf{w}, \mathbf{x}(j))}{\partial x_i(j)} \right\|_2 \right] \lambda^{n-j}, \quad (\text{B.30})$$

where  $n$  is the number of iterations,  $N$  is the number of input nodes,  $(\mathbf{x}(j), \mathbf{t}(j))$  is the sequence of input/output patterns,  $\alpha_j$  is a weighting parameter for each example,  $\mathbf{e}(j)$  is the sequence of errors as in (B.6), p.152, and  $\mathbf{y}(\mathbf{w}, \mathbf{x}(j))$  is the nonlinear output function of the neural network. The solution is derived analogously to the last section from:

$$\begin{aligned} \nabla_{\mathbf{w}_{opt}} \epsilon(n) &= -2 \sum_{j=1}^n \alpha_j \lambda^{n-j} \left[ \mathbf{J}(j) (\mathbf{t}(j) - \mathbf{J}^T(j) \mathbf{w}_{opt} - \xi(j)) \right. \\ &\quad \left. + \sum_{i=1}^N \frac{\partial}{\partial x_i} \left( \frac{\partial \mathbf{y}(\mathbf{w}, \mathbf{x}(j))}{\partial \mathbf{w}} \right) \left( \frac{\partial \mathbf{t}(j)}{\partial x_i} - \frac{\partial \mathbf{y}(\mathbf{w}, \mathbf{x}(j))}{\partial x_i} \right) \right] = 0. \end{aligned} \quad (\text{B.31})$$

With the Jacobi matrix  $\mathbf{J}(j) = \frac{\partial \mathbf{y}(\mathbf{w}, \mathbf{x}(j))}{\partial \mathbf{w}}$  and some rearrangement we get:

$$\hat{\mathbf{w}}(n) = \Phi^{-1}(n) \mathbf{r}(n), \quad (\text{B.32})$$

with:

$$\Phi(n) = \sum_{j=1}^n \alpha_j \mathbf{J}(j) \mathbf{J}^T(j) \lambda^{n-j}, \quad (\text{B.33})$$

$$\mathbf{r}(n) = \sum_{j=1}^n \alpha_j \lambda^{n-j} \left[ \mathbf{J}(j) (\mathbf{t}(j) - \xi(j)) + \sum_{i=1}^N \frac{\partial \mathbf{J}}{\partial x_i} \left( \frac{\partial \mathbf{t}(j)}{\partial x_i} - \frac{\partial \mathbf{y}(\mathbf{w}, \mathbf{x}(j))}{\partial x_i} \right) \right]. \quad (\text{B.34})$$

Rewritten for iterative minimisation gives from (B.33):

$$\Phi(n) = \lambda \Phi(n-1) + \alpha(n) \mathbf{J}(n) \mathbf{J}^T(n), \quad (\text{B.35})$$

and from (B.34):

$$\begin{aligned} \mathbf{r}(n) &= \lambda \mathbf{r}(n-1) + \alpha(n) \mathbf{J}(n) (\mathbf{t}(n) - \xi(n)) \\ &\quad + \sum_{i=1}^N \alpha(n) \frac{\partial \mathbf{J}(n)}{\partial x_i} \left( \frac{\partial \mathbf{t}(n)}{\partial x_i} - \frac{\partial \mathbf{y}(\mathbf{w}, \mathbf{x}(n))}{\partial x_i} \right). \end{aligned} \quad (\text{B.36})$$

Since equation (B.35) is the same as equation (B.12), p.152, in the previous section and only the definition of  $\mathbf{r}(n)$  changed, everything from equation (B.14) up to equation

(B.22) does not change. We will just repeat these equations:

$$\begin{aligned}\Phi^{-1}(n) &= \lambda^{-1} \Phi^{-1}(n-1) - \alpha(n) \lambda^{-2} \Phi^{-1}(n-1) \mathbf{J}(n) \\ &\times \left[ \mathbf{I} + \alpha(n) \lambda^{-1} \mathbf{J}^T(n) \Phi^{-1}(n-1) \mathbf{J}(n) \right]^{-1} \mathbf{J}^T(n) \Phi^{-1}(n-1) .\end{aligned}\quad (\text{B.37})$$

$$\mathbf{K}(n) = \alpha(n) \lambda^{-1} \mathbf{P}(n-1) \mathbf{J}(n) \left[ \mathbf{I} + \alpha(n) \lambda^{-1} \mathbf{J}^T(n) \mathbf{P}(n-1) \mathbf{J}(n) \right]^{-1} . \quad (\text{B.38})$$

$$\mathbf{P}(n) = \lambda^{-1} \mathbf{P}(n-1) - \lambda^{-1} \mathbf{K}(n) \mathbf{J}^T(n) \mathbf{P}(n-1) . \quad (\text{B.39})$$

$$\mathbf{K}(n) = \left[ \alpha(n) \lambda^{-1} \mathbf{P}(n-1) - \alpha(n) \lambda^{-1} \mathbf{K}(n) \mathbf{J}^T(n) \mathbf{P}(n-1) \right] \mathbf{J}(n) . \quad (\text{B.40})$$

$$\mathbf{K}(n) = \alpha(n) \mathbf{P}(n) \mathbf{J}(n) . \quad (\text{B.41})$$

The parameter estimate  $\hat{\mathbf{w}}(n)$  in (B.32) can now be written as:

$$\begin{aligned}\hat{\mathbf{w}}(n) &= \mathbf{P}(n) \mathbf{r}(n) , \\ &= \mathbf{P}(n) \lambda \mathbf{r}(n-1) + \alpha(n) \mathbf{P}(n) \mathbf{J}(n) (\mathbf{t}(n) - \xi(n)) \\ &+ \sum_{i=1}^N \alpha(n) \mathbf{P}(n) \frac{\partial \mathbf{J}(n)}{\partial x_i} \left( \frac{\partial \mathbf{t}(j)}{\partial x_i} - \frac{\partial \mathbf{y}(\mathbf{w}, \mathbf{x}(j))}{\partial x_i} \right) .\end{aligned}\quad (\text{B.42})$$

Substituting (B.39) in the first term of the last equation gives:

$$\begin{aligned}\hat{\mathbf{w}}(n) &= \mathbf{P}(n-1) \mathbf{r}(n-1) - \mathbf{K}(n) \mathbf{J}^T(n) \mathbf{P}(n-1) \mathbf{r}(n-1) \\ &+ \alpha(n) \mathbf{P}(n) \mathbf{J}(n) (\mathbf{t}(n) - \xi(n)) \\ &+ \sum_{i=1}^N \alpha(n) \mathbf{P}(n) \frac{\partial \mathbf{J}(n)}{\partial x_i} \left( \frac{\partial \mathbf{t}(j)}{\partial x_i} - \frac{\partial \mathbf{y}(\mathbf{w}, \mathbf{x}(j))}{\partial x_i} \right) .\end{aligned}\quad (\text{B.43})$$

From (B.32) and (B.41):

$$\begin{aligned}\hat{\mathbf{w}}(n) &= \hat{\mathbf{w}}(n-1) + \mathbf{K}(n) (\mathbf{t}(n) - \mathbf{y}(\hat{\mathbf{w}}(n-1), \mathbf{x}(n))) \\ &+ \sum_{i=1}^N \alpha(n) \mathbf{P}(n) \frac{\partial \mathbf{J}(n)}{\partial x_i} \left( \frac{\partial \mathbf{t}(j)}{\partial x_i} - \frac{\partial \mathbf{y}(\mathbf{w}, \mathbf{x}(j))}{\partial x_i} \right) .\end{aligned}\quad (\text{B.44})$$

The new extended Kalman filter recursions for minimising the error of the output of the network and its derivative are the following equations:

$$\mathbf{K}(n) = \alpha(n) \lambda^{-1} \mathbf{P}(n-1) \mathbf{J}(n) \left[ \mathbf{I} + \lambda^{-1} \mathbf{J}^T(n) \mathbf{P}(n-1) \mathbf{J}(n) \right]^{-1} , \quad (\text{B.45})$$

$$\begin{aligned}\hat{\mathbf{w}}(n) &= \hat{\mathbf{w}}(n-1) + \mathbf{K}(n) (\mathbf{t}(n) - \mathbf{y}(\hat{\mathbf{w}}(n-1), \mathbf{x}(n))) \\ &+ \sum_{i=1}^N \alpha(n) \mathbf{P}(n) \frac{\partial \mathbf{J}(n)}{\partial x_i} \left( \frac{\partial \mathbf{t}(j)}{\partial x_i} - \frac{\partial \mathbf{y}(\mathbf{w}, \mathbf{x}(j))}{\partial x_i} \right) ,\end{aligned}\quad (\text{B.46})$$

$$\mathbf{P}(n) = \lambda^{-1} \mathbf{P}(n-1) - \lambda^{-1} \mathbf{K}(n) \mathbf{J}^T(n) \mathbf{P}(n-1) . \quad (\text{B.47})$$

The formulas for calculating the Jacobi matrix  $\mathbf{J}(n)$ , the derivatives of the Jacobi matrix  $(\partial \mathbf{J} / \partial x_i)$  and the derivatives  $(\partial \mathbf{y} / \partial x_i)$ , i.e. the forces, can be found in Appendix C, p. 157. The matrix  $P$  can be initialised as in the previous section.





---

## Chapter C

### Neural network derivatives

---

For the Kalman filter update equations the computation of the partial derivative of the neural network output function with respect to its parameters, the weights, is necessary. Furthermore, in fitting potential energy surfaces one may want to fit the energy along with the forces, i.e. the derivative of the output with respect to the inputs. In order to derive these derivatives, let us first assume a three-layer network as plotted in Fig. 5.2, p. 44. In the network, the following relationships hold for the first and the second layer respectively, where the input layer is counted as the 0-th layer:

$$\text{Input - hidden layer} \quad \begin{cases} x_j^1(\mathbf{w}, \mathbf{y}^0) = w_{0j}^1 + \sum_i^{n_0} w_{ij}^1 y_i^0 \\ y_j^1(\mathbf{w}, \mathbf{y}^0) = f_j^1(x_j^1) \end{cases}, \quad (\text{C.1})$$

$$\text{Hidden - output layer} \quad \begin{cases} x_k^2(\mathbf{w}, \mathbf{y}^0) = w_{0k}^2 + \sum_j^{n_1} w_{jk}^2 y_j^1 \\ y_k^2(\mathbf{w}, \mathbf{y}^0) = f_k^2(x_k^2) \end{cases}, \quad (\text{C.2})$$

where  $i$  is the index over the  $n_0$  input nodes,  $j$  is the index for the  $n_1$  hidden nodes and  $k$  stands for the  $n_2$  output layer nodes. The output of each neuron depends on the network input vector  $\mathbf{y}^0$  and the current weight vector  $\mathbf{w}$ .

#### C.1 First derivatives with respect to the weights

The partial derivative of the output function  $y_{\tilde{k}}^{N=2}$  in (C.2) with respect to a weight leading into an output node can be calculated using the chain rule:

$$\frac{\partial y_{\tilde{k}}^2}{\partial w_{jk}^2} = \begin{cases} \frac{\partial f_k^2}{\partial x_k^2} \frac{\partial x_k^2}{\partial w_{jk}^2} & : \tilde{k} = k \\ 0 & : \tilde{k} \neq k \end{cases}. \quad (\text{C.3})$$

If the output node in which the weight feeds-in is not the output node of which we want to calculate the partial derivative, the derivative is obviously zero. We define:

$$\delta_{k,k}^2 = \begin{cases} \frac{\partial f_{\tilde{k}}^2}{\partial x_k^2} & : \tilde{k} = k \\ 0 & : \tilde{k} \neq k \end{cases} . \quad (\text{C.4})$$

With the following partial derivative of the output node with respect to the weights from the previous layer and the bias weight respectively

$$\frac{\partial x_k^2}{\partial w_{jk}^2} = \begin{cases} y_j^1 & : j \neq 0 \\ 1 & : j = 0 \end{cases} , \quad (\text{C.5})$$

and (C.4) we can rewrite (C.3) as follows:

$$\frac{\partial y_k^2}{\partial w_{jk}^2} = \begin{cases} \delta_{k,k}^2 y_j^1 & : j \neq 0 \\ \delta_{k,k}^2 & : j = 0 \end{cases} . \quad (\text{C.6})$$

Consequently, the partial derivative of the output function with respect to a weight leading into a hidden node is (with  $n_2$  output nodes):

$$\frac{\partial y_k^2}{\partial w_{ij}^1} = \sum_k^{n_2} \frac{\partial f_k^2}{\partial x_k^2} \frac{\partial x_k^2}{\partial y_j^1} \frac{\partial f_j^1}{\partial x_j^1} \frac{\partial x_j^1}{\partial w_{ij}^1} . \quad (\text{C.7})$$

With the following derivatives:

$$\frac{\partial x_k^2}{\partial y_j^1} = w_{jk}^2 , \quad (\text{C.8})$$

$$\frac{\partial x_j^1}{\partial w_{ij}^1} = \begin{cases} y_i^0 & : i \neq 0 \\ 1 & : i = 0 \end{cases} , \quad (\text{C.9})$$

we get from (C.7):

$$\frac{\partial y_k^2}{\partial w_{ij}^1} = \begin{cases} \frac{\partial f_j^1}{\partial x_j^1} y_i^0 \sum_k^{n_2} \delta_{k,k}^2 w_{jk}^2 & : i \neq 0 \\ \frac{\partial f_j^1}{\partial x_j^1} \sum_k^{n_2} \delta_{k,k}^2 w_{jk}^2 & : i = 0 \end{cases} , \quad (\text{C.10})$$

If we define now (compare with C.4):

$$\delta_{k,j}^1 = \frac{\partial f_j^1}{\partial x_j^1} \sum_k^{n_2} \delta_{k,k}^2 w_{jk}^2 \quad (\text{C.11})$$

we can write, see also (C.6):

$$\frac{\partial y_k^2}{\partial w_{ij}^1} = \begin{cases} \delta_{k,j}^1 y_i^0 & : i \neq 0 \\ \delta_{k,j}^1 & : i = 0 \end{cases} . \quad (\text{C.12})$$

With equations (C.4), (C.11), (C.6) and (C.12) we are now able to extend our results to a neural network with  $N$  layers. We define as above  $\forall \tilde{k} = [1, n_N]$ :

$$\delta_{\tilde{k},j}^p = \begin{cases} \frac{\partial f_{\tilde{k}}^p}{\partial x_j^p} & : \tilde{k} = j \\ 0 & : \tilde{k} \neq j \end{cases} \quad \text{for } p = N, \quad \forall j = [1, n_p],$$

$$\delta_{\tilde{k},j}^p = \frac{\partial f_j^p}{\partial x_j^p} \sum_k^{n_{p+1}} \delta_{\tilde{k},k}^{p+1} w_{jk}^{p+1} \quad \text{for } p \neq N, \quad \forall j = [1, n_p].$$
(C.13)

The first part of equation (C.13) is valid for the output layer only, the second part holds for all other layers. The deltas are defined recursively, i.e. use the first part of eq. (C.13) for  $p = N$  to obtain the delta in the second part for  $p = N - 1$ , use this expression again in the second part but now for  $p = N - 2$ , and so on until one arrives at the output layer. For the calculation of the deltas we need to know the derivative of the activation functions in the net. Given these, we are now able to write the general formula for the derivative of each output node  $y_{\tilde{k}}^N$  with respect to the weight  $w_{ij}^p$ :

$$\frac{\partial y_{\tilde{k}}^N}{\partial w_{ij}^p} = \begin{cases} \delta_{\tilde{k},j}^p y_i^{p-1} & : i \neq 0, \quad \forall p, \tilde{k}, i, j \\ \delta_{\tilde{k},j}^p & : i = 0, \quad \forall p, \tilde{k}, i, j \end{cases},$$
(C.14)

with  $p = [1, N]$ ,  $i = [1, n_{p-1}]$ ,  $j = [1, n_p]$ ,  $\tilde{k} = [1, n_N]$ . The left side of equation (C.14) is defined as the Jacobi matrix, the matrix of the first derivatives:

$$\mathbf{J} = \frac{\partial y_{\tilde{k}}^N}{\partial w_{ij}^p}.$$
(C.15)

## C.2 First derivatives with respect to the inputs

We are now going to derive the partial derivatives of the network outputs with respect to the network inputs. However, we will consider the first derivatives of each nodes output with respect to the input of the network, and not only derivatives of the output layer. Again, let us first assume a three-layer network as plotted in Fig. 5.2, p. 44, with the corresponding equations in (C.1) and (C.2). The derivative of the outputs of the hidden nodes with respect to the inputs is:

$$\frac{\partial y_j^1}{\partial y_i^0} = \frac{\partial f_j^1}{\partial x_j^1} \frac{\partial x_j^1}{\partial y_i^0}.$$
(C.16)

The derivative of the networks output with respect to the inputs is:

$$\frac{\partial y_k^2}{\partial y_i^0} = \sum_{j=1}^{n_1} \frac{\partial f_k^2}{\partial x_k^2} \frac{\partial x_k^2}{\partial y_j^1} \frac{\partial y_j^1}{\partial y_i^0}.$$
(C.17)

Therefore we obtain the following recursions ( $k = [1, n_p]$ ,  $i = [1, n_0]$ ):

$$\frac{\partial y_k^p}{\partial y_i^0} = \frac{\partial f_k^p}{\partial x_k^p} \frac{\partial x_k^p}{\partial y_i^0} \quad p = 1 \quad \forall k, i \quad (\text{input layer}) , \quad (\text{C.18})$$

$$\frac{\partial y_k^p}{\partial y_i^0} = \sum_{j=1}^{n_{p-1}} \frac{\partial f_k^p}{\partial x_k^p} \frac{\partial x_k^p}{\partial y_j^{p-1}} \frac{\partial y_j^{p-1}}{\partial y_i^0} \quad p \geq 2 \quad \forall k, i \quad (\text{hidden, output layer}) . \quad (\text{C.19})$$

With the partial derivatives ( $p = [1, N]$ ,  $j = [1, n_p]$ ,  $i = [1, n_0]$ ):

$$\frac{\partial x_j^p}{\partial y_i^{p-1}} = w_{ij}^p \quad \forall p, j, i , \quad (\text{C.20})$$

we are able to compute the following recursions, with  $k = [1, n_p]$  and  $i = [1, n_0]$ :

$$\boxed{\begin{aligned} \frac{\partial y_k^p}{\partial y_i^0} &= \frac{\partial f_k^p}{\partial x_k^p} w_{ik}^p & p = 1 \quad \forall k, i \quad (\text{input layer}) , \\ \frac{\partial y_k^p}{\partial y_i^0} &= \sum_{j=1}^{n_{p-1}} \frac{\partial f_k^p}{\partial x_k^p} w_{jk}^p \frac{\partial y_j^{p-1}}{\partial y_i^0} & p \geq 2 \quad \forall k, i \quad (\text{hidden, output layer}) . \end{aligned}} \quad (\text{C.21})$$

The equations are again recursively defined, i.e. in order to calculate the forces one has to start with the first part of equation (C.21) for  $p = 1$ , then uses this expression in the second part for  $p = 2$  and then cycles through the latter equation until the output layer is reached. The last obtained partial derivative is then the derivative of the network output with respect to its input, i.e. the forces if the output is the potential and the inputs are the coordinates.

These are the straightforward determined forces. However, for calculating the second derivatives in the next section, we need to define them differently. Let us start with (C.18) but now valid for all layers, with  $p = [1, N]$ ,  $j = [1, n_p]$ ,  $i = [1, n_0]$ :

$$\frac{\partial y_j^p}{\partial y_i^0} = \frac{\partial f_j^p}{\partial x_j^p} \frac{\partial x_j^p}{\partial y_i^0} \quad \forall p, j, i , \quad (\text{C.22})$$

The first term is the derivative of the transfer function and the second term remains to be determined. If we define ( $m = [1, n_p]$ ,  $l = [1, n_0]$ ):

$$\gamma_{ml}^p = \frac{\partial x_m^p}{\partial y_l^0} = \begin{cases} w_{lm}^p & : p = 1 \quad \forall m, l \\ \sum_k w_{km}^p \frac{\partial f_k^p}{\partial x_k^p} \gamma_{kl}^{p-1} & : p \geq 2 \quad \forall m, l \end{cases} , \quad (\text{C.23})$$

we can now rewrite the definition of the partial derivatives of each layers output node with respect to the network inputs, with  $p = [1, N]$ ,  $j = [1, n_p]$ ,  $i = [1, n_0]$ :

$$\boxed{\frac{\partial y_j^p}{\partial y_i^0} = \frac{\partial f_j^p}{\partial x_j^p} \gamma_{ji}^p \quad \forall p, j, i .} \quad (\text{C.24})$$

## C.3 Second derivatives with respect to the weights and inputs

For the minimisation of the forces along with the potential in Appendix B, p. 151, the derivative of the Jacobi matrix with respect to the inputs  $\{x_m\} = \{y_m^0\}$  is required:

$$\frac{\partial \mathbf{J}}{\partial y_m^0} = \frac{\partial^2 y_{\tilde{k}}^N}{\partial y_m^0 \partial w_{ij}^p} . \quad (\text{C.25})$$

The Jacobi matrix is ( $\forall i = [1, n_{p-1}], \forall j = [1, n_p], \forall \tilde{k} = [1, n_N]$ ):

$$\mathbf{J} = \frac{\partial y_{\tilde{k}}^N}{\partial w_{ij}^p} = \begin{cases} \delta_{\tilde{k},j}^p y_i^{p-1} & : i \neq 0 \\ \delta_{\tilde{k},j}^p & : i = 0 \end{cases} . \quad (\text{C.26})$$

We therefore need to calculate:

$$\frac{\partial \mathbf{J}}{\partial y_m^0} = \begin{cases} \frac{\partial}{\partial y_m^0} (\delta_{\tilde{k},j}^p y_i^{p-1}) = y_i^{p-1} \frac{\partial \delta_{\tilde{k},j}^p}{\partial y_m^0} + \delta_{\tilde{k},j}^p \frac{\partial y_i^{p-1}}{\partial y_m^0} & : i \neq 0 \\ \frac{\partial}{\partial y_m^0} (\delta_{\tilde{k},j}^p) & : i = 0 \end{cases} . \quad (\text{C.27})$$

The derivatives of each nodes output with respect to the network inputs  $\partial y_i^{p-1} / \partial y_m^0$  have been already calculated in the recursions of equation (C.21) in the last section. The deltas  $\delta_{\tilde{k},j}^p$  are defined by the recursions in (C.13). The term which is left to be determined is therefore  $\partial \delta_{\tilde{k},j}^p / \partial y_m^0$ . Let us first consider the output layer:

$$\frac{\partial \delta_{\tilde{k},j}^p}{\partial y_m^0} = \frac{\partial}{\partial y_m^0} \begin{cases} \frac{\partial f_{\tilde{k}}^p}{\partial x_j^p} & : \tilde{k} = j, \quad p = N, \quad \forall m, j, \tilde{k} , \\ 0 & : \tilde{k} \neq j, \quad p = N, \quad \forall m, j, \tilde{k} , \end{cases} \quad (\text{C.28})$$

which gives with  $m = [1, n_0], j = [1, n_p], \tilde{k} = [1, n_N]$ :

$$\frac{\partial \delta_{\tilde{k},j}^p}{\partial y_i^0} = \begin{cases} \frac{\partial^2 f_{\tilde{k}}^p}{\partial^2 x_j^p} \frac{\partial x_j^p}{\partial y_i^0} & : \tilde{k} = j, \quad p = N, \quad \forall i, j, \tilde{k} , \\ 0 & : \tilde{k} \neq j, \quad p = N, \quad \forall i, j, \tilde{k} . \end{cases} \quad (\text{C.29})$$

The first part is the second derivative of the transfer function and the second part has been defined previously in equation (C.23):

$$\frac{\partial \delta_{\tilde{k},j}^p}{\partial y_m^0} = \begin{cases} \frac{\partial^2 f_{\tilde{k}}^p}{\partial^2 x_j^p} \gamma_{jm}^p & : \tilde{k} = j, \quad p = N, \quad \forall m, j, \tilde{k} , \\ 0 & : \tilde{k} \neq j, \quad p = N, \quad \forall m, j, \tilde{k} . \end{cases} \quad (\text{C.30})$$

For the hidden layers we get ( $\forall m, j, \tilde{k}$ ):

$$\frac{\partial \delta_{\tilde{k},j}^p}{\partial y_m^0} = \frac{\partial}{\partial y_m^0} \left( \frac{\partial f_j^p}{\partial x_j^p} \sum_k^{n_{p+1}} \delta_{\tilde{k},k}^{p+1} w_{jk}^{p+1} \right) \quad p \neq N , \quad (\text{C.31})$$

$$\frac{\partial \delta_{\tilde{k},j}^p}{\partial y_m^0} = \frac{\partial^2 f_j^p}{\partial^2 x_j^p} \frac{\partial x_j^p}{\partial y_m^0} \sum_k^{n_{p+1}} \delta_{\tilde{k},k}^{p+1} w_{jk}^{p+1} + \frac{\partial f_j^p}{\partial x_j^p} \sum_k^{n_{p+1}} w_{jk}^{p+1} \frac{\partial \delta_{\tilde{k},k}^{p+1}}{\partial y_m^0} \quad p \neq N, \quad (\text{C.32})$$

which leads with the definition of  $\gamma$  in equation (C.23) to:

$$\frac{\partial \delta_{\tilde{k},j}^p}{\partial y_m^0} = \frac{\partial^2 f_j^p}{\partial^2 x_j^p} \gamma_{jm}^p \sum_k^{n_{p+1}} \delta_{\tilde{k},k}^{p+1} w_{jk}^{p+1} + \frac{\partial f_j^p}{\partial x_j^p} \sum_k^{n_{p+1}} w_{jk}^{p+1} \frac{\partial \delta_{\tilde{k},k}^{p+1}}{\partial y_m^0} \quad p \neq N, \quad (\text{C.33})$$

where  $\delta_{\tilde{k},k}^{p+1}/\partial y_m^0$  has been calculated before. We define:

$$\rho_{\tilde{k},j,m}^p = \begin{cases} \frac{\partial^2 f_k^p}{\partial^2 x_j^p} \frac{\partial x_j^p}{\partial y_m^0} & : p = N, \tilde{k} = j, \\ 0 & : p = N, \tilde{k} \neq j, \\ \frac{\partial^2 f_j^p}{\partial^2 x_j^p} \gamma_{jm}^p \sum_k^{n_{p+1}} \delta_{\tilde{k},k}^{p+1} w_{jk}^{p+1} + \frac{\partial f_j^p}{\partial x_j^p} \sum_k^{n_{p+1}} w_{jk}^{p+1} \frac{\partial \delta_{\tilde{k},k}^{p+1}}{\partial y_m^0} & : p \neq N. \end{cases} \quad (\text{C.34})$$

Every term in this equation is now known and therefore the second derivatives in equation (C.27) can be computed, with  $\forall m = [1, n_0], i = [1, n_{p-1}], \forall j = [1, n_p], \forall \tilde{k} = [1, n_N]$ :

$$\frac{\partial \mathbf{J}}{\partial y_m^0} = \frac{\partial^2 y_k^N}{\partial y_m^0 \partial w_{ij}^p} = \begin{cases} y_i^{p-1} \rho_{\tilde{k},j,m}^p + \delta_{\tilde{k},j}^p \frac{\partial y_i^{p-1}}{\partial y_m^0} & : i \neq 0 \\ \frac{\partial \delta_{\tilde{k},j}^p}{\partial y_m^0} & : i = 0 \end{cases} . \quad (\text{C.35})$$

## C.4 Derivatives of the transfer functions

The derivatives of the usually applied transfer functions, the sigmoid and the hyperbolic tangent, can be expressed in terms of the transfer functions themselves.

### Linear function

For the linear transfer function, preferentially used in output nodes, we have:

$$f(x) = x, \quad (\text{C.36})$$

$$\frac{df}{dx} = 1, \quad (\text{C.37})$$

$$\frac{d^2 f}{d^2 x} = 0. \quad (\text{C.38})$$

### Sigmoid function

The sigmoid or logistic function is defined as:

$$f(x) = \frac{1}{1 + e^{-x}} . \quad (\text{C.39})$$

Its first derivative is:

$$\begin{aligned} \frac{df}{dx} &= e^{-x} (1 + e^{-x})^{-2} , \\ &= f(x) (1 - f(x)) . \end{aligned} \quad (\text{C.40})$$

The second derivative is:

$$\frac{d^2f}{dx^2} = f(x) (1 - f(x)) (1 - 2f(x)) . \quad (\text{C.41})$$

### Hyperbolic tangent

The hyperbolic function with two constant scalars  $a$  and  $b$  is:

$$f(x) = a \tanh(b x) . \quad (\text{C.42})$$

The first derivative can be written as:

$$\frac{df}{dx} = \frac{b}{a} (a^2 - f^2(x)) . \quad (\text{C.43})$$

And the second derivative evolves as:

$$\frac{d^2f}{dx^2} = -\frac{2b^2}{a^2} f(x) (a^2 - f^2(x)) . \quad (\text{C.44})$$

Recommended values for the parameters are  $a=1.7159$  and  $b=\frac{2}{3}$ . When this function is used with transformed inputs, the variance of the output will be close to 1 because the effective gain of the function, the first derivative around  $x=0$ , is roughly 1. Furthermore,  $f(\pm 1) = \pm 1$  and the second derivative has its extremes at  $x = 1$ , i.e. when using it in an output node one should scale the target outputs to the range  $\pm 1$ .

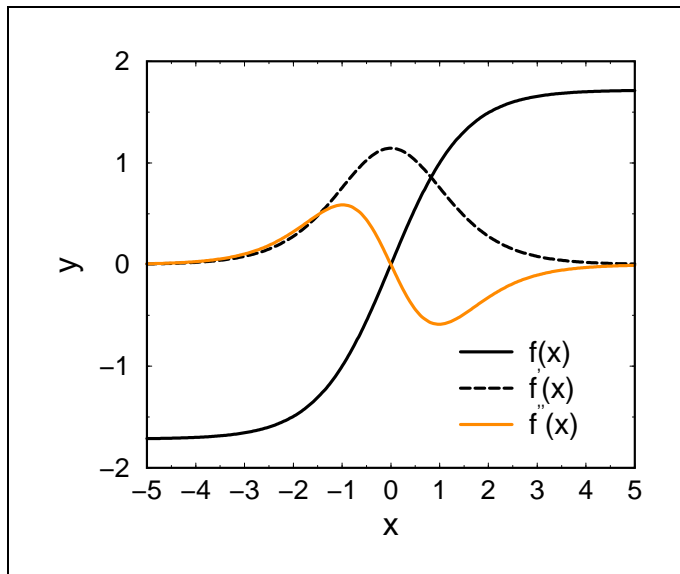


Figure C.1: Activation function  $f(x) = 1.7159 \tanh(\frac{2}{3}x)$  and its first and second derivatives.





---

## Chapter D

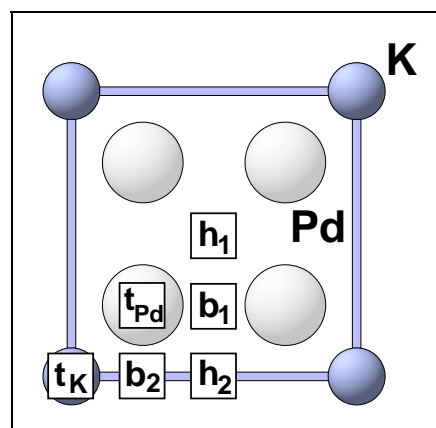
### Sampling of the *ab initio* PES for $\text{H}_2/(2 \times 2)\text{K}/\text{Pd}(100)$

---

In this appendix we present the sampling of the points in the two-dimensional cuts for the six-dimensional *ab initio* PES of hydrogen dissociation on the  $\text{K}(2 \times 2)/\text{Pd}(100)$  surface. The displayed two dimensions are the distance of the molecule to the surface and the H-H distance. The *ab initio* cuts have been calculated with the FP-LAPW method as implemented in the WIEN 97-code [82]. We have discussed the PES in detail in Sec. 10.2. The black dots in the following graphs correspond to the *ab initio* total energies. For visualisation these points have been interpolated in two dimensions with the Fortran routine *LOTPS* from *CMLIB* which performs a smooth interpolation of scattered data by local thin plate splines [151]. The gridded output of *LOTPS* is then plotted with the software package *xfarbe* [152].

The different adsorption sites are illustrated in Fig. D.1. The  $(2 \times 2)$  potassium covered  $\text{Pd}(100)$  surface exhibits two inequivalent hollow sites  $h_1$ ,  $h_2$ , two bridge sites  $b_1$ ,  $b_2$  and two top sites  $t_{\text{Pd}}$ ,  $t_{\text{K}}$  above a palladium and a potassium atom. Eleven configurations of the  $\text{H}_2$  molecule above these sites have been calculated. Overall, we have calculated 659 *ab initio* total energies. The majority, i.e. 575 points, were sampled within the eleven elbow plots. Additionally, we determined 84 energies associated with the corrugation of the energy barriers across the unit cell as discussed in Sec. 10.2, which we will not repeat here.

To distinguish between the different orientations we will characterise them by the position of the two H-atoms and the centre of mass of the molecule. For instance, a geometry  $h_1\text{-}b_1\text{-}h_2$  denotes the situation where the centre of mass is over the bridge site  $b_1$  and the H-atoms are oriented towards the hollow sites  $h_1$  and  $h_2$ , respectively.



**Figure D.1:** Surface geometry of the  $(2 \times 2)$  potassium covered  $\text{Pd}(100)$  surface with two inequivalent hollow sites  $h_1$ ,  $h_2$ , two bridge sites  $b_1$ ,  $b_2$  and two top sites  $t_{\text{Pd}}$ ,  $t_{\text{K}}$  above a palladium and a potassium atom.

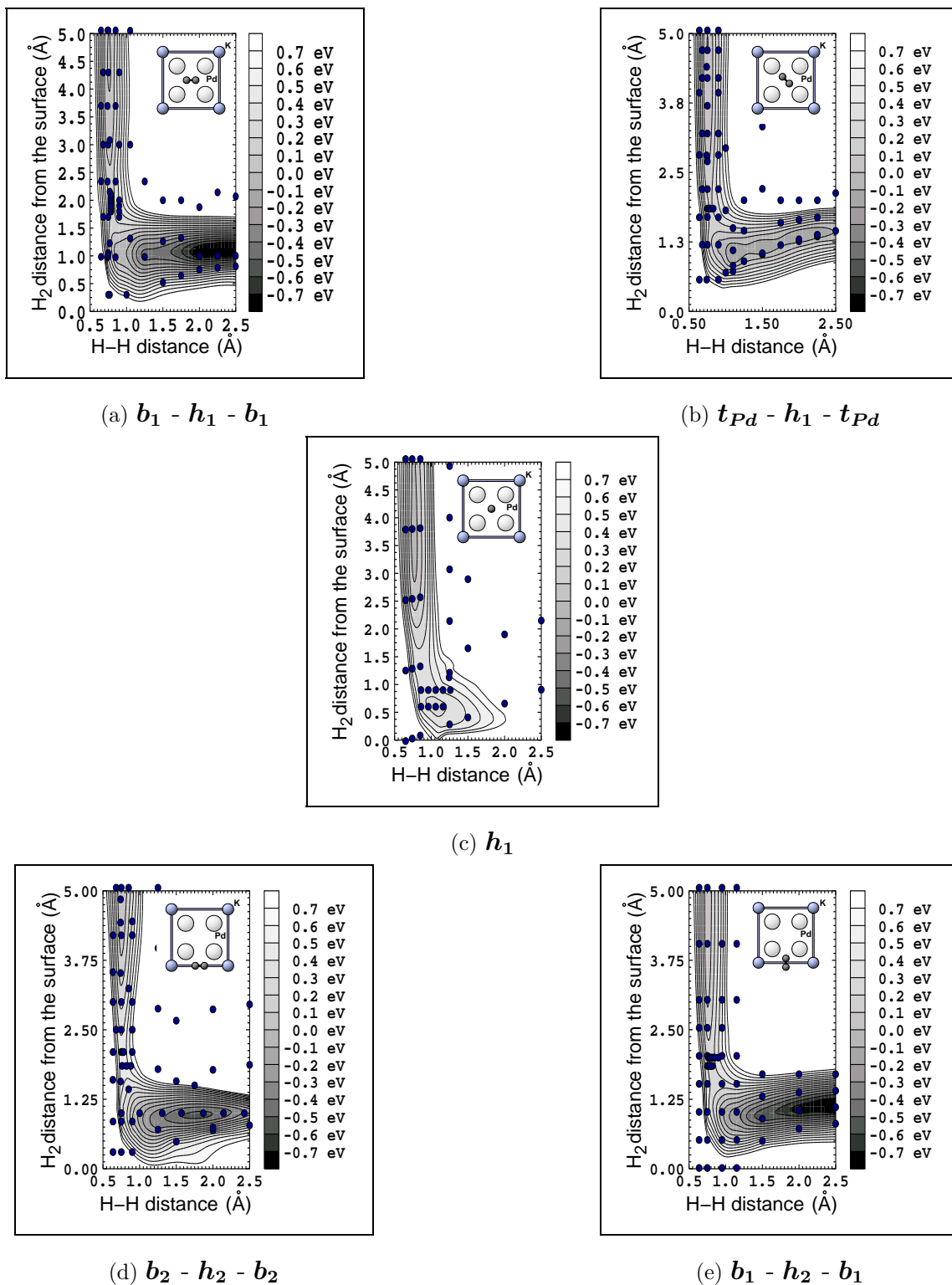


Figure D.2: 2D contour plots of the PES of  $H_2$  dissociation on the  $K(2 \times 2)/Pd(100)$  surface for different geometries. The black dots indicate the *ab initio* calculated energies. (a)+(b)+(c):  $H_2$  above the hollow site  $h_1$ . (d)+(e):  $H_2$  above the hollow site  $h_2$ .

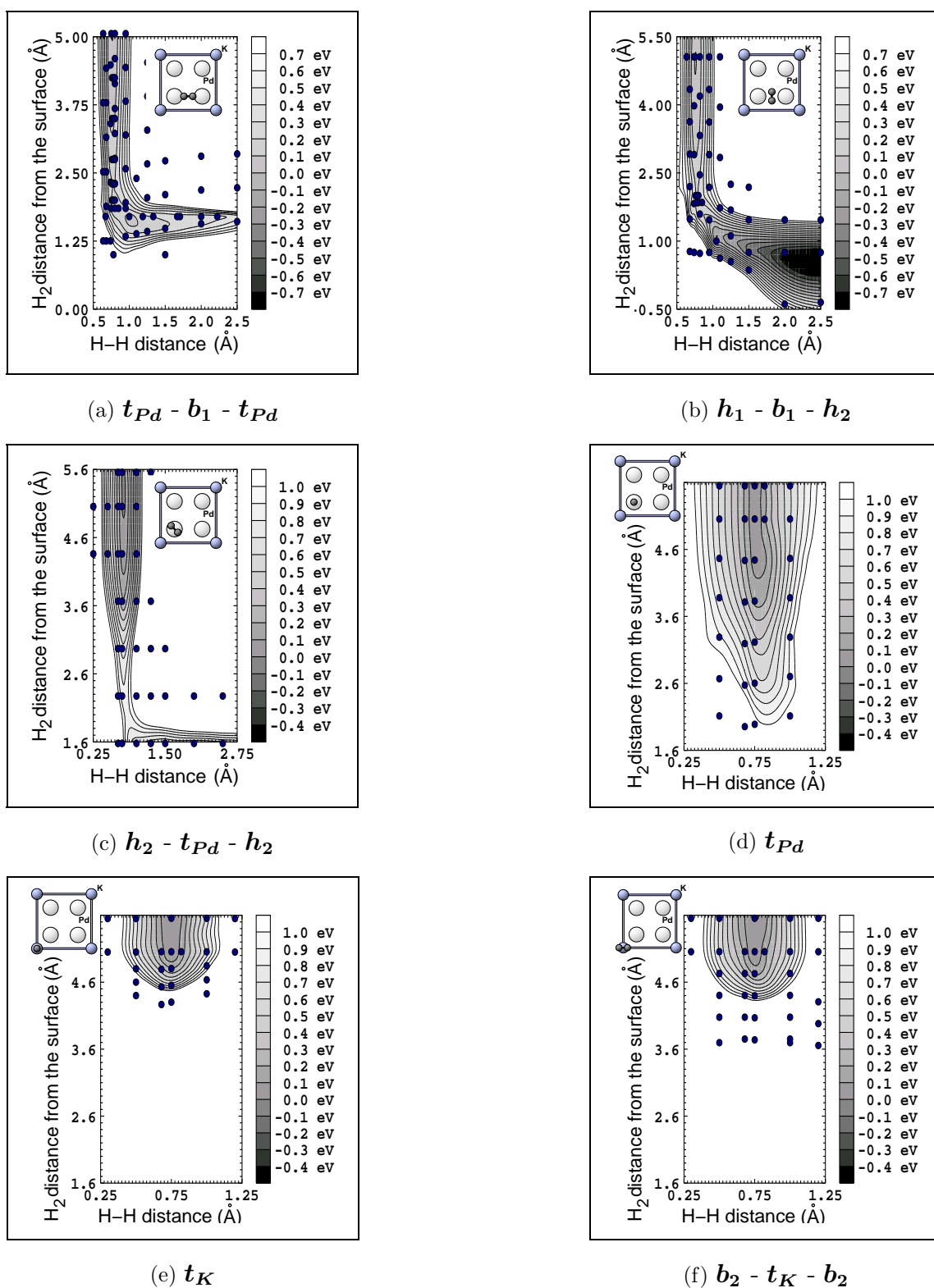


Figure D.3: 2D contour plots of the PES of  $H_2$  dissociation on the  $K(2 \times 2)/Pd(100)$  surface for different geometries. The black dots indicate the ab initio calculated energies. (a)+(b):  $H_2$  above the bridge site  $b_1$ . (c)+(d):  $H_2$  on top Pd. (e)+(f):  $H_2$  on top K.



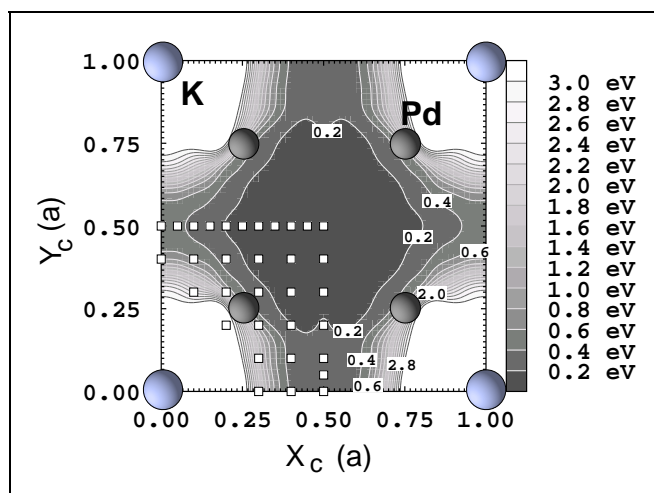
## Chapter E

### Neural Network PES for $\text{H}_2/(2 \times 2)\text{K}/\text{Pd}(100)$

The *ab initio* PES for the dissociation of  $\text{H}_2$  on a potassium covered Pd(100) surface has been interpolated with neural networks. The 659 *ab initio* energies have been divided into a training set of 619 points and a test set containing 40 energies. In addition to the *ab initio* energies we used 60 energies representing the vacuum region of the molecule in the training set. Far away from the surface the molecule is in the vacuum, i.e. the energy does not depend on the distance of the centre of mass from the surface. In order to include this in the fit we repeated the *ab initio* calculated vacuum energy at  $Z = 5.05 \text{ \AA}$  for values up to  $Z = 15 \text{ \AA}$ . A 8–24–18–1 sl neural network, i.e. with two hidden layers and 685 weights, has been used for the interpolation. The Kalman Filter parameters were  $\lambda(0) = 0.98$ ,  $\lambda_0 = 0.99903$  and the adaptive threshold parameter was 0.1. The training error on the 679 examples measured 0.021 eV after 5000 epochs, the test error of 0.074 eV was also within the desired *ab initio* accuracy.

The following cuts should be compared to the contour plots of the previous appendix. However, we emphasise that the elbow plots presented here have been obtained from the *six-dimensional* neural network interpolation of the *ab initio* data. In contrast, the cuts of App. D show *two-dimensional* interpolations of the *ab initio* energies.

Figure E.1: *Potential energy as a function of the lateral coordinates  $X_c$  and  $Y_c$ . The molecule is oriented parallel to the surface at a height  $z = 1.9 \text{ \AA}$  with a bond length  $d = 0.77 \text{ \AA}$  and angles  $\phi = 0^\circ$  and  $\theta = 90^\circ$  as plotted in Fig 10.8, p. 122. The contour spacing is 0.2 eV. The white squares mark the calculated *ab initio* points. a: length of the  $(2 \times 2)$  cell.*



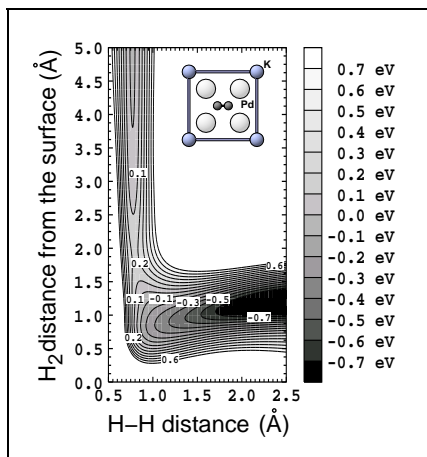
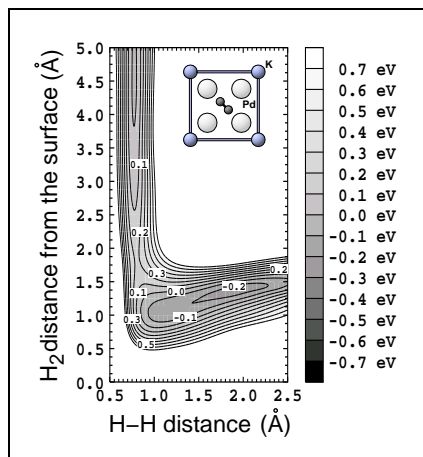
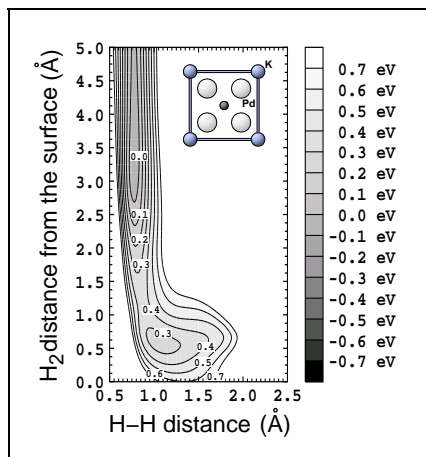
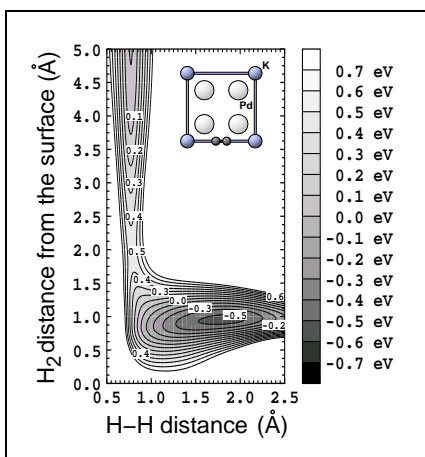
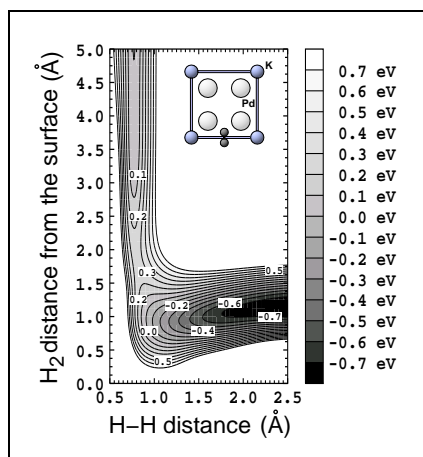
(a)  $b_1 - h_1 - b_1$ (b)  $t_{Pd} - h_1 - t_{Pd}$ (c)  $h_1$ (d)  $b_2 - h_2 - b_2$ (e)  $b_1 - h_2 - b_1$ 

Figure E.2: 2D cuts through the 6D NN-PES of hydrogen dissociation on the  $\text{K}(2 \times 2)/\text{Pd}(100)$  surface. (a)+(b)+(c):  $\text{H}_2$  above the hollow site  $h_1$ . (d)+(e):  $\text{H}_2$  above the hollow site  $h_2$ .

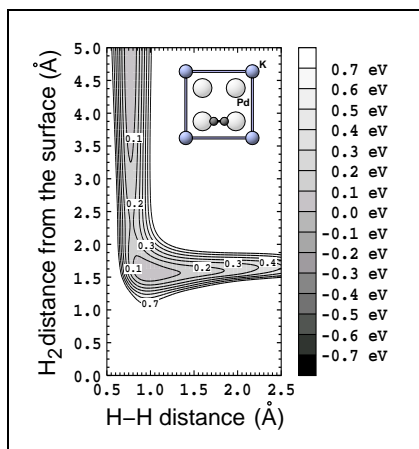
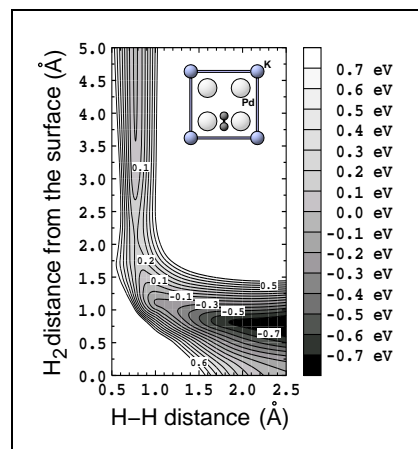
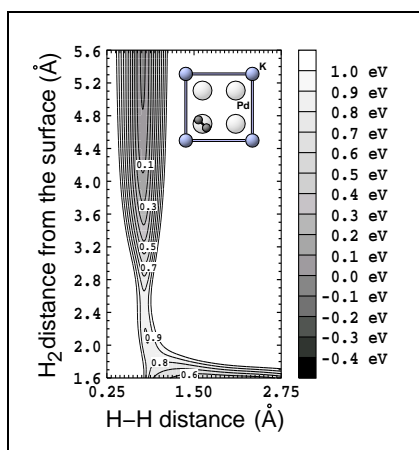
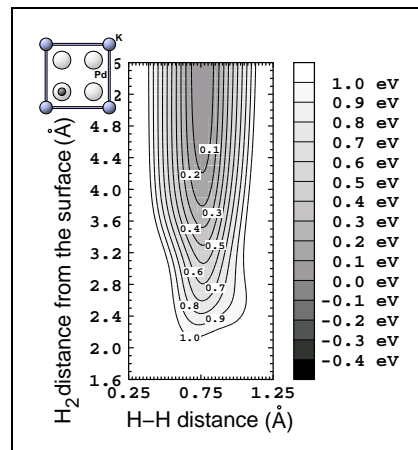
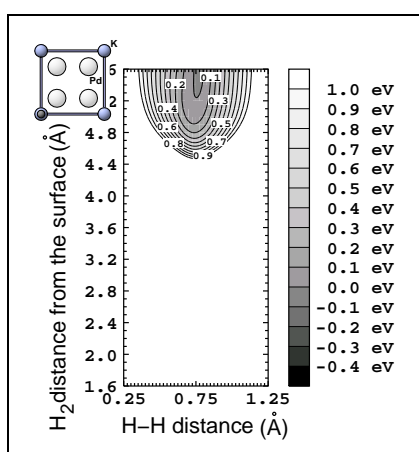
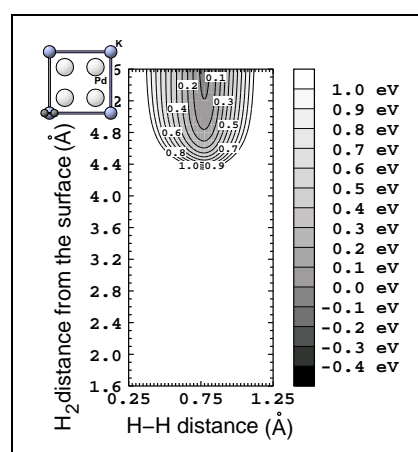
(a)  $t_{Pd} - b_1 - t_{Pd}$ (b)  $h_1 - b_1 - h_2$ (c)  $h_2 - t_{Pd} - h_2$ (d)  $t_{Pd}$ (e)  $t_K$ (f)  $b_2 - t_K - b_2$ 

Figure E.3: 2D cuts through the 6D NN-PES of hydrogen dissociation on the  $K(2 \times 2)/Pd(100)$  surface. (a)+(b):  $H_2$  above the bridge site  $b_1$ . (c)+(d):  $H_2$  on top Pd. (e)+(f):  $H_2$  on top K.





---

## Chapter F

### Abbreviations

---

AGEKF	Adaptive Global Extended Kalman Filter
APW	Augmented Plane Waves
BP	Backpropagation
CG	Conjugate Gradients
EKF	Extended Kalman Filter
FP-LAPW	Full Potential - Linear Augmented Plane Wave method
GGA	Generalised Gradient Approximation
LAPW	Linear Augmented Plane Waves
LDA	Local Density Approximation
MD	Molecular Dynamics
NN	Neural Network
PES	Potential Energy Surface
RMSE	Root Mean Squared Error
TDS	Thermal desorption spectroscopy



---

## Bibliography

---

- [1] G. A. Somorjai, *Introduction to Surface Chemistry and Catalysis* (Wiley, New York, 1994). 5, 22
- [2] A. Zangwill, *Physics at Surfaces* (Cambridge University Press, Cambridge, 1992). 5, 15
- [3] C. Stampfl and M. Scheffler, *Surf. Sci.* **433**, 119 (1999). 5
- [4] K. Rendulic, G. Anger and A. Winkler, *Surf. Sci.* **208**, 404 (1989). 5, 6, 16, 17, 22, 24, 91, 104, 105, 108
- [5] J. M. Thomas and W. J. Thomas, *Principles and Practice of Heterogenous Catalysis* (VCH, Weinheim, 1997). 5
- [6] C. Stampfl, M. V. Ganduglia-Pirovano, K. Reuter and M. Scheffler, to appear in *Surf. Sci.* **500**, (2001). 5
- [7] P. Kratzer and M. Scheffler, *Computing in Science & Engineering*, subm. (2001). 5
- [8] P. Blaha, K. Schwarz, P. Sorantin and S. B. Trickey, *Comp. Phys. Comm.* **59**, 399 (1990). 5, 28, 29
- [9] M. C. Payne, M. P. Teter, D. C. Allan, T. A. Arias and J. D. Joannopoulos, *Rev. Mod. Phys.* **64**, 1045 (1992). 5, 28
- [10] G. Kresse and J. Furthmüller, *Phys. Rev. B* **54**, 11169 (1996). 5, 28
- [11] M. Bockstedte, A. Kley, J. Neugebauer and M. Scheffler, *Comp. Phys. Comm.* **107**, 187 (1997). 5, 28, 61
- [12] J. W. Clark and M. L. Ristig, *Scientific Applications of Neural Nets* (Springer Verlag, Heidelberg, 1998). 5, 41
- [13] J. Zupan and J. Gasteiger, *Neural Networks for Chemists* (VCH Verlag, Weinheim, 1993). 5, 41
- [14] D. H. Zanette and A. S. Mikhailov, *Phys. Rev. E* **58**, 872 (1998). 5, 41
- [15] G. J. Kroes, *Prog. Surf. Sci.* **60**, 1 (1999). 6, 7
- [16] A. Groß, *Surf. Sci. Rep.* **32**, 291 (1998). 6, 13, 20, 21
- [17] B. Hammer, K. W. Jacobsen and J. K. Nørskov, *Phys. Rev. Lett.* **69**, 1971 (1992). 6

- [18] B. Hammer, M. Scheffler, K. W. Jacobsen and J. K. Nørskov, Phys. Rev. Lett. **73**, 1400 (1994). [6](#), [14](#), [17](#), [18](#), [27](#), [34](#), [35](#)
- [19] J. A. White, D. M. Bird, M. C. Payne and I. Stich, Phys. Rev. Lett. **73**, 1404 (1994). [6](#), [17](#), [34](#), [35](#)
- [20] B. Hammer and M. Scheffler, Phys. Rev. Lett. **74**, 3487 (1995). [6](#), [18](#), [125](#)
- [21] J. A. White, D. M. Bird and M. C. Payne, Phys. Rev. B **53**, 1667 (1996). [6](#), [34](#), [35](#)
- [22] S. Wilke and M. Scheffler, Phys. Rev. B **53**, 4926 (1996). [6](#), [14](#), [17](#), [34](#), [35](#), [36](#), [74](#), [83](#), [113](#), [117](#), [118](#), [119](#), [125](#)
- [23] G. Wisenekker, G. J. Kroes and E. J. Baerends, J. Chem. Phys. **104**, 7344 (1996). [6](#), [34](#), [35](#)
- [24] A. Eichler, G. Kresse and J. Hafner, Phys. Rev. Lett. **77**, 1119 (1996). [6](#), [34](#), [35](#)
- [25] W. Dong and J. Hafner, Phys. Rev. B **56**, 15396 (1996). [6](#), [34](#), [35](#)
- [26] A. Groß, B. Hammer, M. Scheffler and W. Brenig, Phys. Rev. Lett. **73**, 3121 (1994). [6](#), [8](#), [73](#)
- [27] G. Kresse, Phys. Rev. B **62**, 8295 (2000). [6](#), [8](#), [17](#), [36](#), [98](#)
- [28] H. F. Busnengo, W. Dong and A. Salin, Chem. Phys. Lett. **320**, 328 (2000). [6](#), [8](#), [17](#), [36](#), [98](#)
- [29] S. Wilke and M. Scheffler, Surf. Sci. **329**, L605 (1995). [6](#), [14](#), [17](#), [34](#), [35](#), [36](#), [74](#), [83](#), [91](#), [92](#), [105](#), [114](#), [116](#), [119](#), [123](#)
- [30] S. Wilke and M. Scheffler, Phys. Rev. Lett. **76**, 3380 (1996). [6](#), [91](#), [92](#), [116](#), [119](#), [123](#)
- [31] J. J. Mortensen, B. Hammer and J. K. Nørskov, Phys. Rev. Lett. **80**, 4333 (1998). [6](#), [22](#)
- [32] F. Besenbacher, I. Chorkendorff, B. S. Clausen, B. Hammer, A. M. Molenbroek, J. K. Nørskov and I. Stensgard, Science **279**, 1913 (1998). [6](#)
- [33] C. M. Wei, A. Groß and M. Scheffler, Phys. Rev. B **57**, 15572 (1998). [6](#), [8](#), [9](#), [22](#), [23](#), [91](#), [92](#), [104](#), [106](#), [108](#), [114](#), [115](#), [116](#), [119](#), [121](#), [123](#), [124](#), [125](#)
- [34] F. Solymosi and I. Kovacs, Surf. Sci. **260**, 139 (1992). [6](#), [24](#), [111](#), [112](#), [114](#)
- [35] J. K. Nørskov, S. Holloway and N. D. Lang, Surf. Sci. **137**, 65 (1984). [6](#)
- [36] D. M. Bird, Faraday Discuss. **110**, 335 (1998). [6](#), [22](#), [125](#)
- [37] S. Wilke and M. H. Cohen, Surf. Sci. **380**, L446 (1997). [6](#), [24](#), [91](#), [111](#), [112](#), [113](#), [114](#), [115](#), [116](#), [117](#), [125](#)

- [38] M. R. Sorensen and A. F. Voter, *J. Chem. Phys.* **112**, 9599 (2000). 6
- [39] A. Groß, M. Bockstedte and M. Scheffler, *Phys. Rev. Lett.* **79**, 701 (1997). 7, 61
- [40] A. Groß and M. Scheffler, *Phys. Rev. B* **57**, 2493 (1998). 7, 17, 61, 62
- [41] A. Groß, S. Wilke and M. Scheffler, *Phys. Rev. Lett.* **75**, 2718 (1995). 7, 8, 9, 14, 19, 36, 37, 62, 67, 73, 74, 83
- [42] J. H. McCreery and G. W. Jr., *J. Chem. Phys.* **63**, 2340 (1975). 8, 36
- [43] A. Forni, G. Wiesenekker, E. J. Baerends and G. F. Tantardini, *J. Phys.: Condens. Matter* **7**, 7195 (1995). 8, 36
- [44] A. Forni and G. F. Tantardini, *Surf. Sci.* **352-354**, 142 (1996). 8, 36
- [45] U. Kleinekathöfer, K. T. Tang, J. P. Toennies and C. L. Yiu, *J. Chem. Phys.* **111**, 3377 (1999). 8, 36
- [46] D. E. Makarov and H. Metiu, *J. Chem. Phys.* **108**, 590 (1998). 8, 36
- [47] A. Groß, M. Scheffler, M. J. Mehl and D. A. Papaconstatopoulos, *Phys. Rev. Lett.* **82**, 1209 (1999). 8, 36, 38
- [48] H. F. Busnengo, W. Dong and A. Salin, *J. Chem. Phys.* **112**, 7641 (2000). 8, 17, 36, 98
- [49] G. Wiesenekker, G. J. Kroes and E. J. Baerends, *J. Chem. Phys.* **104**, 7344 (1996). 8
- [50] J. Dai and J. Light, *J. Chem. Phys.* **107**, 1676 (1997). 8
- [51] A. Groß, C. M. Wei and M. Scheffler, *Surf. Sci. Lett.* **416**, 1095 (1998). 8, 23, 24, 36, 62, 73, 96, 104, 105, 108, 123, 133
- [52] J. Hertz, A. Krogh and R. G. Palmer, *Introduction to the Theory of Neural Computation* (Addison-Wesley, Reading, 1996). 8, 41, 56, 80
- [53] R. Rojas, *Theorie der neuronalen Netze* (Springer Verlag, Heidelberg, 1996). 8, 41
- [54] Y. LeCun, L. Bottou, G. B. Orr and K. R. Müller, in *Neural Networks: Tricks of the Trade*, G. B. Orr and K. R. Müller (eds.) (Springer Verlag, Heidelberg, 1998). 8, 41, 46, 51, 56, 57
- [55] G. Cybenko, *Mathematics of Control, Signals, and Systems* **2**, 303 (1989). 8, 45
- [56] K. Hornik, M. Stinchcombe and H. White, *Neural Networks* **2**, 359 (1989). 8, 45
- [57] T. B. Blank, S. D. Brown, A. W. Calhoun and D. J. Doren, *J. Chem. Phys.* **103**, 4129 (1995). 8, 36, 41, 42, 78

- [58] D. F. R. Brown, M. N. Gibbs and D. C. Clary, *J. Chem. Phys.* **105**, 7597 (1996).  
8, 41
- [59] K. T. No, B. H. Chang, S. Y. Kim, M. S. Jhon and H. A. Scheraga, *Chem. Phys. Lett.* **271**, 152 (1997). 8, 41
- [60] H. Gassner, M. Probst, A. Lauenstein and K. Hermansson, *J. Phys. Chem. A* **102**, 4596 (1998). 8, 41
- [61] F. V. Prudente and J. J. S. Neto, *Chem. Phys. Lett.* **287**, 585 (1998). 8, 41
- [62] A. Groß and M. Scheffler, *Phys. Rev. B* **61**, 8425 (2000). 8, 104, 105, 133
- [63] G. Kroes, E. Baerends and R. Mowrey, *Phys. Rev. Lett.* **78**, 3583 (1997). 8, 17, 62
- [64] G. Kroes, E. Baerends and R. Mowrey, *J. Chem. Phys.* **107**, 3309 (1997). 8, 62
- [65] A. Groß, *Appl. Phys. A* **67**, 627 (1998). 13
- [66] G. Eilmsteiner, W. Walkner and A. Winkler, *Surf. Sci.* **352**, 263 (1996). 16
- [67] P. Bratu, W. Brenig, A. Gross, M. Hartmann, U. Höfer, P. Kratzer and R. Russ, *Phys. Rev. B* **54**, 5978 (1996). 16
- [68] O. Hinrichsen, F. Rosowski, A. Hornung, M. Muhler and G. Ertl, *J. Catal.* **165**, 33 (1997). 16
- [69] C. Rettner, H. Michelsen and D. Auerbach, *J. Chem. Phys.* **102**, 4625 (1995). 16
- [70] M. Scheffler and C. Stampfl, *Theory of Adsorption on Metal Substrates; in: Handbook of Surface Science, Vol. 2: Electronic Structure, Eds. K. Horn and M. Scheffler* (Elsevier, Amsterdam, 1999). 17, 19
- [71] M. Beult, M. Riedler and K. D. Rendulic, *Chem. Phys. Lett.* **247**, 249 (1995). 17
- [72] M. Kay, G. Darling, S. Holloway, J. White and D. Bird, *Chem. Phys. Lett.* **245**, 311 (1995). 17
- [73] A. Eichler, J. Hafner, A. Groß and M. Scheffler, *Phys. Rev. B* **59**, 13297 (1999). 18, 20
- [74] H. Hjelmberg, B. I. Lundquist and J. K. Nørskov, *Phys. Scr.* **20**, 192 (1979). 18
- [75] K. Rendulic and A. Winkler, *Surf. Sci.* **299/300**, 261 (1994). 20, 22
- [76] G. Ertl, S. B. Lee and M. Weiss, *Surf. Sci.* **111**, (1981). 22
- [77] Z. Y. Li, R. N. Lamb, W. Allison and R. F. Willis, *Surf. Sci.* **211-212**, 931 (1989). 22

- [78] E. Bertel, P. Roos and J. Lehmann, Phys. Rev. B **52**, (1995). 22
- [79] J. K. Brown, A. C. Luntz and P. A. Schultz, J. Chem. Phys. **95**, 3767 (1991). 22
- [80] B. E. Hayden and C. L. A. Lamont, Faraday Discuss. Chem. Soc. **91**, 415 (1991). 22
- [81] C. Resch, V. Zhukov, A. Lugstein, H. F. Berger, A. Winkler and K. D. Rendulic, Chem. Phys. **177**, 421 (1993). 22
- [82] P. Blaha, K. Schwarz and J. Luitz, *WIEN97, A Full Potential Linearized Plane Wave Package for Calculating Crystal Properties* (K. Schwarz, Techn. Universität Wien, Austria, 1999), ISBN 3-9501031-0-4. 25, 112, 165
- [83] B. Kohler, S. Wilke, M. Scheffler, R. Kouba and C. Ambrosch-Draxl, Comp. Phys. Comm. **94**, 31 (1996). 25, 31, 112
- [84] M. Petersen, F. Wagner, L. Hufnagel, M. Scheffler, P. Blaha and K. Schwarz, Comp. Phys. Commun. **126**, 294 (2000). 25
- [85] M. Born and J. R. Oppenheimer, Ann. Phys. **84**, 457 (1927). 25
- [86] P. Hohenberg and W. Kohn, Phys. Rev. **136**, B864 (1964). 26
- [87] W. Kohn and L. J. Sham, Phys. Rev. **140**, A1133 (1965). 26
- [88] J. P. Perdew and A. Zunger, Phys. Rev. B **23**, 5048 (1981). 27
- [89] A. D. Becke, Phys. Rev. A **38**, 3098 (1988). 27
- [90] C. Lee, W. Yang and R. G. Parr, Phys. Rev. B **37**, 785 (1988). 27
- [91] J. P. Perdew, J. A. Chevary, S. H. Vosko, K. A. Jackson, M. R. Pederson, D. J. Singh and C. Fiolhais, Phys. Rev. B **46**, 6671 (1992). 27, 112, 117
- [92] J. P. Perdew, K. Burke and M. Ernzerhoff, Phys. Rev. Lett. **77**, 3865 (1996). 27
- [93] B. Hammer, L. B. Hansen and J. K. Nørskov, Phys. Rev. B **59**, 7413 (1999). 27
- [94] B. Hammer, L. B. Hansen and J. K. Nørskov, Phys. Rev. B **59**, 7413 (1999). 27
- [95] O. K. Andersen, Phys. Rev. B **12**, 3060 (1975). 28
- [96] H. L. Skriver, *The LMTO Method* (Springer-Verlag, Berlin, 1984). 28
- [97] S. Y. Savrasov and D. Y. Savrasov, Phys. Rev. B **46**, 12181 (1992). 28
- [98] D. J. Singh, *Planewaves, Pseudopotentials and the LAPW Method* (Kluwer Academic, Boston, 1994). 28, 29
- [99] D. R. Hamann, M. Schlüter and C. Chiang, Phys. Rev. Lett. **43**, 1494 (1979). 28

- [100] J. C. Slater, Phys. Rev. **51**, 846 (1937). 29
- [101] J. C. Slater, Advances in Quantum Chemistry **1**, 35 (1964). 29
- [102] H. Bross, Phys. Kondens. Mater. **3**, 119 (1964). 29
- [103] H. Bross, Z. Phys. B **81**, 233 (1990). 29
- [104] D. Singh, Phys. Rev. B **43**, 6388 (1991). 30
- [105] R. P. Feynmann, Phys. Rev. **56**, 340 (1939). 31
- [106] P. Pulay, Mol. Phys. **17**, 197 (1969). 31
- [107] M. Scheffler, J. P. Vigneron and G. B. Bachelet, Phys. Rev. B **31**, 6541 (1985). 31
- [108] F. Bloch, Z. Phys. **52**, 555 (1928). 31
- [109] J. D. Pack and H. J. Monkhorst, Phys. Rev. B **16**, 1748 (1977). 33
- [110] W. Brening and H. Kasai, Surf. Sci. **213**, 170 (1989). 37, 74
- [111] R. E. Cohen, M. J. Mehl and D. A. Papaconstantopoulos, Phys. Rev. B **50**, 14694 (1994). 38
- [112] M. J. Mehl and D. A. Papaconstantopoulos, Phys. Rev. B **54**, 4519 (1996). 38
- [113] M. J. Mehl and D. A. Papaconstantopoulos, *Topics in Computational Materials Science, Ch. 5* (World Scientific, Singapore, 1998). 38
- [114] J. C. Slater and G. F. Koster, Phys. Rev. **94**, 1498 (1954). 38
- [115] D. Ormoneit, Neural Networks **12**, 1405 (1999). 41
- [116] O. Maas, J. P. Boulanger and S. Thiria, Neurocomputing **30**, 53 (2000). 41
- [117] F. Scarselli and A. C. Tsoi, Neural Networks **12**, 15 (1998). 41
- [118] J. G. Attali and G. Pages, Neural Networks **10**, 1069 (1997). 41
- [119] C. Kiesling, *Contributed paper for the Lepton-Photon Conference* (Conference Proceedings, Beijing, 1995). 42
- [120] W. S. McCulloch and W. Pitts, Bulletin of Mathematical Biophysics **5**, 115 (1943). 42
- [121] G. Cybenko, *Continuous Valued Nerual Networks with Two Hidden Layers Are Sufficient* (Technical Report, Department of Computer Science, Tufts University, Medford, MA, 1988). 45
- [122] S. V. Chakravarthy and J. Ghosh, Neural Networks **10**, 459 (1997). 46



- [123] M. J. L. Orr, *Neural Computation* **7**, 606 (1995). [46](#)
- [124] W. H. Press, B. P. Flannery, S. A. Teukolsky and W. T. Vetterling, *Numerical Recipes* (Cambridge University Press, Cambridge, 1989). [47](#), [49](#), [50](#), [51](#), [63](#), [64](#)
- [125] A. Gelb, *Applied Optimal Estimation* (MIT Press, Cambridge, MA, 1974). [51](#)
- [126] S. Shah, F. Palmieri and M. Datum, *Neural Networks* **5**, 779 (1992). [51](#), [151](#)
- [127] T. B. Blank and S. D. Brown, *J. Chemometrics* **8**, 391 (1994). [51](#), [52](#), [54](#), [77](#), [78](#), [82](#)
- [128] D. P. Bertsekas, *SIAM J. Optimization* **6**, 807 (1996). [51](#)
- [129] S. Haykin, P. Yee and E. Derbez, *IEEE Trans. Signal Processing* **45**, 2774 (1997). [53](#)
- [130] S. Haykin, A. H. Sayed, J. Zeidler, P. Yee and P. Wei, *IEEE Trans. Signal Processing* **45**, 1118 (1997). [53](#)
- [131] T. S. Rönkvallsson, in *Neural Networks: Tricks of the Trade*, G. B. Orr and K. R. Müller (eds.) (Springer Verlag, Heidelberg, 1998). [56](#)
- [132] T. Plate, in *Neural Networks: Tricks of the Trade*, G. B. Orr and K. R. Müller (eds.) (Springer Verlag, Heidelberg, 1998). [56](#)
- [133] J. P. F. Sum, C. S. Leung and L. W. Chan, *Technical report CS-TR-96-05* (Department of Computer Science and Engineering, Chinese University of Hong Kong, 1996). [56](#)
- [134] P. v. d. Smagt and G. Hirzinger, in *Neural Networks: Tricks of the Trade*, G. B. Orr and K. R. Müller (eds.) (Springer Verlag, Heidelberg, 1998). [58](#), [82](#)
- [135] R. Car and M. Parinello, *Phys. Rev. Lett.* **55**, 2471 (1985). [61](#)
- [136] A. Groß and M. Scheffler, *J. Vac. Sci. Technol. A* **15**, 1624 (1997). [62](#)
- [137] C. Rettner and D. Auerbach, *Phys. Rev. Lett.* **77**, 404 (1996). [62](#)
- [138] A. Groß and M. Scheffler, *Phys. Rev. Lett.* **77**, 405 (1996). [62](#)
- [139] G. W. Flake, in *Neural Networks: Tricks of the Trade*, G. B. Orr and K. R. Müller (eds.) (Springer Verlag, Heidelberg, 1998). [79](#)
- [140] M. L. Burke and R. J. Madix, *Surf. Sci.* **237**, 1 (1990). [105](#)
- [141] A. Berkó and F. Solymosi, *Surf. Sci.* **187**, 359 (1987). [112](#), [113](#)
- [142] J. Burchhardt, E. Lundgren, M. M. Nielsen, J. N. Andersen and D. L. Adams, *Surf. Rev. and Lett.* **3**, 1339 (1996). [112](#)

- [143] C. Resch, H. F. Berger, K. D. Rendulic and E. Bertel, Surf. Sci. **316**, 1105 (1994).  
112
- [144] M. Karikorpi, S. Holloway, N. Henriksen and J. K. Nørskov, Surf. Sci. **179**, (1987).  
132, 133
- [145] I. Beauport, K. Al-Shamery and H.-J. Freund, Chem. Phys. Lett. **256**, 641 (1996).  
137, 138
- [146] T. Klüner, H.-J. Freund, V. Staemmler and R. Kosloff, Phys. Rev. Lett. **80**, 5208  
(1998). 138
- [147] S. Thiel, *Dissertation* (Freie Universität, Berlin, 2000). 138, 140
- [148] M. Pykavy, *Dissertation* (Ruhr-Universität, Bochum, 1999). 138, 140
- [149] M. Pykavy, V. Staemmler, O. Seiferth, H. Kuhlenbeck and H.-J. Freund, Surf.  
Sci., subm. (2000) . 138, 140
- [150] M. Pykavy, T. Klüner, H.-J. Freund and V. Staemmler, Surf. Sci., subm. (2001)  
. 138
- [151] R. Franke, *Naval Postgraduate School* (CMLIB, <http://gams.nist.gov/>, 1981).  
165
- [152] A. Preusser, *xfarbe - Fill Area with Bicubics* (Fritz-Haber-Institut der MPG,  
Berlin, <http://www.fhi-berlin.mpg.de/grz/pub/>, 1999). 165

---

*I would like to thank ...*

---

- ... Matthias Scheffler for introducing me to a very interesting subject, giving me the opportunity to present my results at a number of conferences and also the chance to do my Ph.D. in a city like Berlin.
- ... then of course Axel Groß for his kind supervision of my thesis, even after he moved to Munich, his patience, the very helpful discussions and for letting me use some of his programs. Furthermore I am also grateful for his never-vanishing humour, the invitations to a couple of conferences and the accompanied evenings at different Bavarian beer gardens and even the Oktoberfest - which was quite something for a northerner like me -, his ability to mix Caipirinhas and Daniela's and Axel's generous hospitality!
- ... Doug Doren for the interaction on the field of Neural Networks. Doug, it was nice to have you and your family here in Berlin.
- ... Stefan and Thorsten for the interesting collaboration.
- ... Veronica's and Max's help with the LAPW code. Special thanks go to Max for introducing me to Berlin's nightlife and Drum & Bass, connected to some extremely pleasant evenings.
- ... Karsten for reviewing my thesis, for some very helpful comments on that and the interesting discussions we had - not only at the institute.
- ... My roommates Rossitza, Tosja and Sixten for a nice time. I would also like to thank Osvaldo for establishing "the Club". Furthermore, thanks to Tini for showing me the wild, wild east.
- ... Gaby for the help with all sorts of things.
- ... and last but not least, I would like to thank my dearest Jana-Maria for her willingness to come with me to Berlin and simply for the wonderful time.



---

## Curriculum Vitae

---

### Personal Data

Name Sönke Lorenz  
Address Fritz-Haber-Institut der Max-Planck-Gesellschaft  
14195 Berlin, Germany  
Date of Birth 7/17/68 in Rendsburg, Germany

### Education

Since 01/97 Ph.D. Student, Fritz-Haber-Institut der MPG, Berlin  
Research Subject : Reactions on Surfaces with Neural Networks  
10/93 - 09/96 Christian Albrechts Universität in Kiel; Major: Physics  
09/10/96 Diploma, Diploma Thesis: Photoemission Final States with  
realistic Surface Potentials - a Multigrid Method.  
09/92 - 08/93 University of London, King's College, Physics  
08/01/93 Bachelor of Science, Final Project: Photoluminescence and  
Absorption in InP/InGaAs Quantum Well Structures  
10/90 - 09/92 Christian Albrechts Universität in Kiel; Major: Physics  
09/21/92 Pre-Diploma  
10/90 - 09/92 University of Hagen; Major: Economics

### Employment and Related Experience

April 2000 Delegate of the Free University, Berlin, at the Conference  
"National Model United Nations (NMUN) 2000" in New York  
09/90 - 12/96 News Reader, News Editor at "R.SH", a Radiostation in Kiel  
10/93 - 07/96 Student Assistant, Dep. of Theoretical Physics, University of Kiel  
Tutor for Mechanics, Thermodynamics and Quantum Mechanics  
07/94 - 09/94 Siemens, Central Research Unit in Munich  
Internship in the Surface-Wave Devices Group  
07/88 - 06/90 Military Service

### Fellowship

01/97 - 12/99 Max-Planck-Fellowship for Ph.D Students

