Representing high-dimensional potential-energy surfaces for reactions at surfaces by neural networks

Sönke Lorenz,[1] Axel Groß,[2] and Matthias Scheffler[1]

[1] *Fritz-Haber-Institut der Max-Planck-Gesellschaft, Faradayweg 4-6, D-14195 Berlin-Dahlem, Germany*

[2] *Physik-Department T30, Technische Universität München, D-85747 Garching, Germany*

(Dated: July 22, 2004)

The determination of dissociative adsorption probabilities based on first-principles total-energy calculations requires a numerically efficient and accurate interpolation scheme in order to be able to run a sufficient number of trajectories. Here we present a neural network scheme for the construction of a continuous potential energy surface (PES). We illustrate the accuracy and efficiency of our method for H_2 interacting with the (2×2) potassium covered Pd(100) surface. The sticking probability of H_2/K(2×2)/Pd(100) is determined by molecular dynamics simulations on the neural network PES and compared to results using an independent analytical interpolation.

Keywords: Computer simulations, density functional calculations, molecular dynamics, sticking, hydrogen, palladium, low index single crystal surface

I. INTRODUCTION

The interaction of molecules with surfaces is of great technological and fundamental relevance for, e.g. heterogeneous catalysis, growth of semiconductor devices, corrosion, and wear protection [1]. The key quantity in the description of this interaction is the potential-energy surface (PES). For the reaction of simple molecules with surfaces, PESs can nowadays be mapped out in great detail by first-principles electronic structure calculations [2–4]. Still, the determination of reaction probabilities requires many molecular-dynamics simulations (treating the nuclei either as classical or as quantum-mechanical particles) [5–7]. For such calculations a continuous and differentiable representation of the PES is needed, and this must be build from a discrete set of total-energy data that can be obtained from electronic structure calculations. Of course, also direct *ab initio* molecular dynamics simulations are possible [8]. However, in such calculations many electronic-structure calculations are repeated again and again for practically the same atomic configuration, and as such computations are very demanding, i.e. costly, it is rather time-consuming to calculate a sufficient number of trajectories in order to obtain an accurate statistics in the determination of the reaction probabilities [5, 9].

Therefore a numerically efficient and accurate representation of the PES is crucial as a second step, after the PES is mapped out on a mesh, in order to have a computationally efficient scheme for the evaluation of potential energies and gradients. Several methods have been suggested in the past for the description of, e.g., the dissociation of molecules on surfaces. Analytical expansions using symmetry-adapted functions [10, 11] are computationally very efficient, but they are rather inflexible and their complexity grows rapidly with increasing dimensionality of the considered systems. The notion that most of the corrugation in a molecule-surface PES is embedded in the atom-surface interaction leaves a relatively smooth molecular interpolation function to be adjusted [12, 13]. Interpolation schemes based on this idea have been successfully applied to the study of the interaction dynamics of molecular hydrogen with metal surfaces [13–15]. It still remains to be seen whether this method can be extended to higher dimensional problems. In the modified Shepard interpolation scheme [16, 17], the potential energy in the vicinity of an *ab initio* input point is expanded in a second-order Taylor series. This avoids the introduction of a regular grid so that the accuracy of the PES can be iteratively improved by a sampling scheme based on classical trajectory calculations.

As an alternative, we propose the use of multilayer, feedforward neural networks [18] for the continuous representation of a PES. 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. Neural networks have already been applied to problems involving function approximation in general [19] and more recently to the represention of PESs [20, 21]. These studies dealt with a few configurational parameters. We extend these applications of the neural network approach to the high-dimensional description of reactions of molecules on surfaces. The resulting neural network PES is smooth, continuous and leads to a relatively simple but accurate expression for the PES. The evaluation of the potential with a fitted neural network is cheap and the derivatives of the potential energy are easily obtainable which makes neural networks suitable, and in fact efficient, for molecular dynamics calculations.

Here we present and analyse the neural network PES that we built from the *ab initio* results of H_2/K(2×2)/Pd(100). We will demonstrate that neural networks can be successfully applied for the determination of dissociative adsorption probabilities, but we will also discuss problems in the construction of a neural network PES. The number of parameters and required data points for an accurate neural network fit can be rather smale but only if an appropriate set of input points in

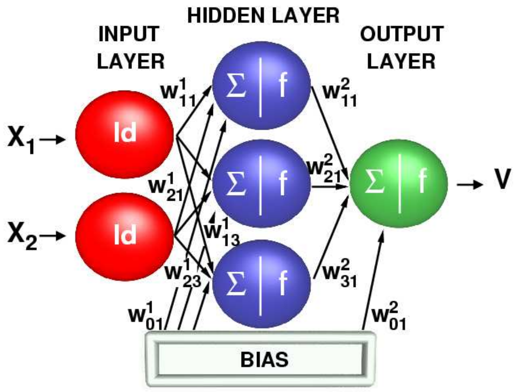

FIG. 1: Schematic architecture of an artificial neural network, here a multilayer feed-forward net. The neurons are arranged in layers. The bias unit acts as an adjustable offset.

chosen and if *a priori* knowledge about the considered system such as underlying symmetries is properly taken into account.

II. TECHNICAL DETAILS

An (artificial) neural network is a highly flexible, non-linear model that can in principle approximate any continuous function to arbitrary accuracy [22]. 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. 1. 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 the outside world. 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.*

The input variables of the neural network are here the coordinates of the six degrees of freedom of the hydrogen molecule. The neural network output function for a neural network with one hidden layer and one output node representing the potential energy is an analytical function which can be written as:

$$V_{nn}(\mathbf{x}) \;=\; f_2\left(w_{01}^2 \;+\; \sum_j w_{j1}^2 \; f_1\left(w_{0j}^1 + \sum_{i=1}^{6} w_{ij}^1 x_i \right)\right), \tag{1}$$

where $V_{nn}(\mathbf{x})$ is the potential energy, $\mathbf{x} = (x_1, \ldots, x_6)$ are the six coordinates of the H_2 molecule, f_1 and f_2 are non-linear functions and the $\{w_{ij}^p\}$ are the parameters of the representation, the so-called weights, here from layer p, node i to layer $p+1$, node j. The bias weights such as $\{w_{0j}^1\}$ act as an adjustable offset of the activation function. The output function is a nested function f of f. The non-linear basis functions (often called activation functions) give neural networks their non-linear capabilities. The function must be differentiable for the optimization of the parameters. We have tested different non-linear functions and found that Fermi-like functions are well suited.

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. 1 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.

When fitting the neural network parameters, the weights are optimized by comparing the output with known correct answers. The cost function is normally taken as the sum of the squared residuals between the true or targeted energies $V(\mathbf{x})$ obtained from *ab initio* total-energy calculations and the actual output of the network $V_{nn}(\mathbf{x})$:

$$E\left(\{\mathbf{x}\}\right) = \sum_{i=1}^{n} \; u_i \left(V_{nn}(\mathbf{x}^{(i)}) - V(\mathbf{x}^{(i)}) \right)^2 / \sum_{i=1}^{n} \; u_i \tag{2}$$

where n is the number of examples in the data set and the u_i are weighting factors (see below). The optimization of the network weights is performed by some iterative optimization scheme until a desirable quality, measured by the root mean squared error (RMSE) which is the square root of $E\left(\{\mathbf{x}\}\right)$, is reached. In order to minimize the RMSE, the network cycles repeatedly through the following steps of the optimization process: (1) present the network one example of the data set, (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 minimize the cost function, and last, if the mean squared error reaches a desired lower bound, stop the iteration, otherwise go to (1). However, as it will be discussed below, it may be advisable if not necessary to add more information on the physics of the system.

The neural network expression (1) can be viewed as a fitting function. The advantage of using a neural network expression is that one can build on the detailed experience already gained in developing efficient algorithm for finding the optimum fit parameters, a process that in the language of neural networks is called "learning". The optimization of the weights is usually done by gradient-based learning methods like steepest decent, conjugate gradients or quasi-Newton algorithms. We use the global, extended Kalman filter [23]. Detailed aspects of our approach for the neural network fit are

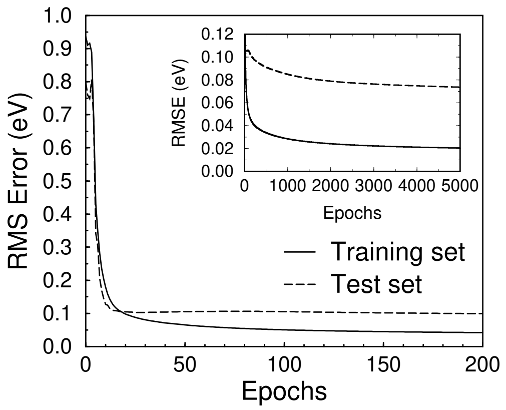

FIG. 2: The training and test root mean square error as a function of the number of epochs for the neural network fit of the *ab initio* energies of the system H_2/K(2×2)/Pd(100). The inset shows the errors for a larger number of epochs.

described elsewhere [24]. We build a neural network PES from density-functional theory (DFT) total-energy data. The Kohn-Sham equations were solved using the full-potential linear augmented plane wave (FP-LAPW) method [25, 26] together with the generalized gradient approximation (GGA) [27] for the exchange-correlation functional.

III. RESULTS AND DISCUSSION

As an example of molecular dynamics simulations based on a neural network PES that was derived from *ab initio* calculations we present the determination of the dissociative adsorption probability of H_2 on the (2×2) potassium covered Pd(100) surface. Overall, we have calculated 659 DFT total energies for the construction of the neural network PES. The majority were sampled within eleven two-dimensional cuts as a function of the distance Z_c of the molecular center-of-mass from the surface and the intra-molecular bond length d above five different high-symmetry sites. Additionally, we focused on the energies associated with the corrugation of the energy barriers across the unit cell. We did so by placing the molecule in the most favorable configuration for dissociation, i.e. parallel to the surface at a height and a bond length which corresponds to the region where the hydrogen bond breaks. We then scanned the corrugation by moving the molecule across the unit cell in such a configuration. The calculated PES of H_2/K(2×2)/Pd(100) is in fact strongly corrugated. The dissociation of H_2 at K(2×2)/Pd(100) is activated with the minimum barrier of 0.18 eV above the hollow site, whereas the dissociation barrier at the ontop position above the potassium atom is at least 5 eV [24].

In order to avoid the use of prohibitively many input points, an efficient and reliable selection scheme for the input data is asked for. Unfortunately, there is no *a priori* criterion for such a selection. In activated systems, the regions of the PES close to the minimum barriers are certainly the most important for the determination of reaction probabilities. These regions should be accurately reproduced which can be achieved by a finer grid of input points and/or higher weighting factors for the error at these points. If dynamical effects such as steering [28] or dynamic trapping [14, 29] play a decisive role in the reaction dynamics, then also regions of the PES further away from the barrier positions become crucial in the fit. Therefore we propose a feedback between dynamical simulations and the sampling of input points in order to obtain a reliable fit, as for example also used in [16]. This means that molecular dynamics simulations should be performed on the interpolated PES already during the fitting procedure. The regions of the configuration space which are most often visited during the dynamics runs are the most relevant for the reaction dynamics. These regions should then be fitted with a higher accuracy than regions that are less often visited.

For the neural network representation of *ab initio* PES for the system H_2/K(2×2)/Pd(100) we divided the 659 *ab initio* energies into a training set of 619 and a test set of 40 energies. The selection of the *ab initio* points was guided by our experience gained in the neural network fitting of the H_2/S(2×2)/Pd(100) PES [24] which exhibits similar features as the H_2/K(2×2)/Pd(100) PES. Furthermore, it is advantageous to include all *a priori* knowledge about the considered system such as the underlying symmetries. Thus it is ensured that during optimisation of the Neural Network the emphasize lies on fitting the crucial chemical process, the bond-breaking of the molecule, and no fitting power is wasted on fitting the known symmetry of the surface. We have achieved this by choosing eight symmetry-adapted coordinates as the input to the neural network fit [24] which are

$$
\begin{aligned}
x_1 &= d \ , \\
x_2 &= e^{Z_c/2} \ , \\
x_3 &= \sin^2\theta \cos 2\phi \left[\cos GX_c - \cos GY_c\right] e^{Z_c/2} \ , \\
x_4 &= \sin^2\theta \cos 2\phi \left[\cos 2GX_c - \cos 2GY_c\right] e^{Z_c/2} \ , \\
x_5 &= \cos^2\theta \ e^{Z_c/2} \ , \\
x_6 &= \left[\cos GX_c + \cos GY_c\right] e^{Z_c/2} \ , \\
x_7 &= \left[\cos 2GX_c + \cos 2GY_c\right] e^{Z_c/2} \ , \\
x_8 &= \sin^4\theta \cos 4\phi \left[\cos 2GX_c + \cos 2GY_c\right] e^{Z_c/2} \ . \qquad (3)
\end{aligned}
$$

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 weighting factor $e^{-Z_c/2}$ is used in order to ensure that the energy in the vacuum only depends on the bond length. The neural network performs a *non-linear fit* on these new inputs. The transformation needs to be done only once per surface symmetry. It is important to note that the terms $x_1, \cdots, x_8$ represent only the minimum symmetry employed in the data set, any other

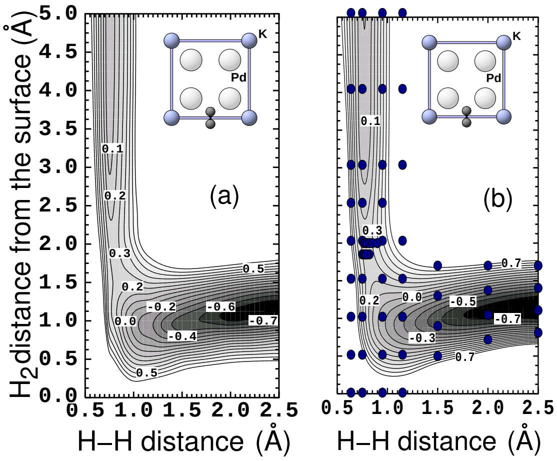

FIG. 3: Two-dimensional cuts through the six-dimensional PES of H_2/K(2×2)/Pd(100). The molecular configuration within the surface unit cell is depicted in the insets. a) two-dimensional spline interpolation of *ab initio* results; b) two-dimensional cut through a six-dimensional neural network interpolation. The dots in (b) mark the calculated grid of *ab initio* energies for this specific cut.

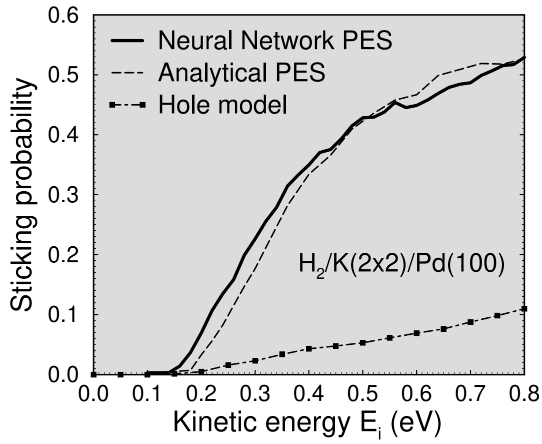

FIG. 4: Sticking probability versus kinetic energy of the dissociation of H_2 on a K(2×2)/Pd(100) using a neural network and an analytical interpolation of the *ab initio* PES. The statistical error of the sticking probabilities is ±2.2 %. In addition, the results according to the hole model [30] (see text) are shown.

symmetry can then be learned by the Neural Network.

Without such an approach, the number of training points necessary for an accurate fit becomes much larger. A similar effect can also be achieved by including the gradients of the potential in the optimization procedure. A 8−24−18−1sl neural network, i.e. with two hidden layers, each with 24 or 18 nodes respectively, and 685 weights, a sigmoidal transfer function in the hidden layers and a linear function in the output layer was used for the fit. In an activated system such as H_2/K(2×2)/Pd(100) the configurations close to the valley of each elbow plot are most crucial for the reaction dynamics, as revealed by molecular dynamics simulations. During the fit, for these configurations the weighting factors u_i in the cost function Eq. (2) were chosen to be ten times larger than for other regions where the potential energy is significantly elevated. Thus the region accessible for molecular dynamics simulations is represented with a higher accuracy.

The training and test root mean squared errors for the neural network fit as a function of the number of so-called epochs are shown in Fig. 2. An epoch corresponds to the process of presenting the network one whole cycle of the data. The training and test errors drop very quickly 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 another 30 meV. The training error after 5000 epochs measured 21 meV, the test error on the 40 energies not presented during the optimization of the network weights was 74 meV. Both errors lie well below the desired accuracy of 0.1 eV.

Figure 3 illustrates the accuracy of the neural network fit. One two-dimensional cut through the six-dimensional potential energy surface, a so-called elbow plot, obtained by the DFT calculations (Fig. 3a) is compared to the corresponding neural network elbow plot (Fig. 3b). This cut corresponds to the dissociation above the hollow site between two K atoms which is hindered by a barrier 0.27 eV. Both cuts are based on the DFT-points indicated as dots in Fig. 3b. However, the DFT-cut in Fig. 3a shows a two-dimensional spline interpolation of these points, whereas Fig. 3b is a two-dimensional cut through a six-dimensional neural network interpolation. It is obvious that a finer grid of input points was used close to the minimum barrier for an accurate representation of this region.

A more detailed analysis of the distribution of the prediction error for the training and test set reveals that 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 and errors are less significant. This region has been fitted with a smaller weight during the minimization 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. 30 points of the test set exhibit an error smaller than 0.05 eV and 8 between 0.05 eV and 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.

Based on the neural network PES, we 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. the statistical error of the sticking probability is $1/\sqrt{2000} \approx 2.2\,\%$. The sticking curve is shown in Fig. 4. In addition, the sticking probability in the classical sudden approximation or the so-called "hole model" [30] is shown. It corresponds to the integrated barrier distribution $P_b(E)$ which is the fraction of the two-dimensional cuts through the six-dimensional configuration space, as shown in Fig. 3 for which the barrier towards dissociation is less than E. The large difference between the hole model and the calculated sticking probability demonstrates that there are strong dynamical effects in the dissociation dynamics of H_2/K(2×2)/Pd(100). Although the dissociative adsorption is activated, there is also significant steering of the impinging molecules to low-barrier configurations, as has already been found for the similar system H_2/S(2×2)/Pd(100) [5, 31, 32].

In order to check the reliability of the neural network fit, we obtained an independent analytical fit of the H_2/K(2×2)/Pd(100) PES using the same parametrization scheme as employed for H_2/S(2×2)/Pd(100) [32]. The adsorption probability based on this analytical PES with 1000 trajectories per energy is also plotted in Fig. 4. Both dynamical results agree well which lends further credibility to the neural network fit. However, we emphasise that the neural network approach to the representation of *ab initio* data is *general*, i.e. it is not restricted to the application of dissociation problems only. Furthermore, neural networks are flexible and can easily be adjusted to higher dimensions, in contrast to the analytical representation which is not immediately transferable to other problems and whose extension to higher dimensions is rather problematic.

IV. CONCLUSIONS

Using the hydrogen dissociation on the (2×2) potassium-covered Pd(100) surface as an example, we have demonstrated that neural networks provide an accurate, flexible and efficient scheme for the continuous representation of an *ab initio* potential energy surface. Taking into account the symmetries underlying the particular system and the feedback from dynamical simulations, a relatively moderate number of training points is needed in order to obtain a reliable fit. The evaluation of the potential and its gradient with a fitted neural network is cheap. Thus the neural network representation of *ab initio* energies with molecular dynamics simulations provides a efficient tool to study reaction processes where extensive statistics are required.

[1] C. Stampfl, M. V. Ganduglia-Pirovano, K. Reuter, and M. Scheffler, Surf. Sci. **500** (2002) 368.
[2] B. Hammer, M. Scheffler, K. Jacobsen, and J. Nørskov, Phys. Rev. Lett. **73** (1994) 1400.
[3] S. Wilke and M. Scheffler, Phys. Rev. B **53** (1996) 4926.
[4] A. Groß, Surf. Sci. **500** (2002) 347.
[5] A. Groß, Surf. Sci. Rep. **32** (1998) 291.
[6] G.-J. Kroes, Prog. Surf. Sci. **60** (1999) 1.
[7] G.-J. Kroes, A. Groß, E. J. Baerends, M. Scheffler, and D. A. McCormack, Acc. Chem. Res. **35** (2002) 193.
[8] M. E. Tuckerman, J. Phys.: Condens. Matter **14** (2003) R1297.
[9] A. Groß, M. Bockstedte, and M. Scheffler, Phys. Rev. Lett. **79** (1997) 701.
[10] G. Wiesenekker, G.-J. Kroes, and E. J. Baerends, J. Chem. Phys. **104** (1996) 7344.
[11] A. Groß and M. Scheffler, Phys. Rev. B **57** (1998) 2493.
[12] H. F. Busnengo, A. Salin, and W. Dong, J. Chem. Phys. **112** (2000) 7641.
[13] G. Kresse, Phys. Rev. B **62** (2000) 8295.
[14] H. F. Busnengo, W. Dong, and A. Salin, Chem. Phys. Lett. **320** (2000) 328.
[15] M. A. Di Cesare, H. F. Busnengo, W. Dong, and A. Salin, J. Chem. Phys. **118** (2003) 11226.
[16] C. Crespos, M. A. Collins, E. Pijper, and G.-J. Kroes, Chem. Phys. Lett. **376** (2003) 566.
[17] C. Crespos, M. A. Collins, E. Pijper, and G.-J. Kroes, J. Chem. Phys. **120** (2004) 2392.
[18] J. Hertz, A. Krogh, and R. G. Palmer, *Introduction to the Theory of Neural Computation*, Addison-Wesley, Reading, 1996.
[19] F. Scarselli and A. C. Tsoi, Neural Networks **11** (1998) 15.
[20] T. B. Blank, A. W. Brown, S. D. Calhoun, and D. J. Doren, J. Chem. Phys. **103** (1995) 4129.
[21] D. F. R. Brown, M. N. Gibbs, and D. C. Clary, J. Chem. Phys. **105** (1996) 7597.
[22] K. Hornik, M. Stinchcombe, and H. White, Neural Networks **2** (1989) 359.
[23] S. Shah, F. Palmieri, and M. Datum, Neural Networks **5** (1992) 779.
[24] S. Lorenz, A. Groß, and M. Scheffler, to be published.
[25] B. Kohler, S. Wilke, M. Scheffler, R. Kouba, and C. Ambrosch-Draxl, Comput. Phys. Commun. **94** (1996) 31.
[26] P. Blaha, K. Schwarz, and J. Luitz, *Wien97*, Technische Universität Wien, Austria, 1999, ISBN 3-9501031-0-4.
[27] 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** (1992) 6671.
[28] A. Groß, S. Wilke, and M. Scheffler, Phys. Rev. Lett. **75** (1995) 2718.
[29] A. Groß and M. Scheffler, J. Vac. Sci. Technol. A **15** (1997) 1624.
[30] M. Karikorpi, S. Holloway, N. Henriksen, and J. K. Nørskov, Surf. Sci. **179** (1987) L41.
[31] A. Groß, C.-M. Wei, and M. Scheffler, Surf. Sci. **416** (1998) L1095.
[32] A. Groß and M. Scheffler, Phys. Rev. B **61** (2000) 8425.