LEED I/V determination of the structure of a MoO_3 monolayer on Au(111): testing the performance of the CMA-ES evolutionary strategy algorithm, differential evolution, a genetic algorithm and tensor LEED based structural optimization

E. Primorac[a], H. Kuhlenbeck[a,*], H.-J. Freund[a]

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

Abstract

The structure of a thin MoO_3 layer on Au(111) with a c(4×2) superstructure was studied with LEED I/V analysis. As proposed previously (Quek et al, Surf. Sci. 577 (2005) L71), the atomic structure of the layer is similar to that of a MoO_3 single layer as found in regular α-MoO_3. The layer on Au(111) has a glide plane parallel to the short unit vector of the c(4×2) unit cell and the molybdenum atoms are bridge-bonded to two surface gold atoms with the structure of the gold surface being slightly distorted. The structural refinement of the structure was performed with the CMA-ES evolutionary strategy algorithm which could reach a Pendry R-factor of ~0.044. In the second part the performance of CMA-ES is compared with that of the differential evolution method, a genetic algorithm and the Powell optimization algorithm employing I/V curves calculated with tensor LEED.

Keywords: MoO_3/Au(111), I/V LEED, CMA-ES, tensor LEED, genetic algorithm, differential evolution

*Corresponding author

Email address: kuhlenbeck@fhi-berlin.mpg.de (H. Kuhlenbeck)

Introduction

Molybdenum is extensively used as a key component in mixed oxide catalysts like for instance iron-molybdate catalysts for the selective oxidation of methanol to formaldehyde [1]. It is assumed that Mo sites play a relevant role for the reactivity of these systems, but many questions regarding the active sites and the reaction mechanism are still open at present [1]. Pure MoO_3 exhibits good activity for several reactions like the partial oxidation of methanol to formaldehyde [2] and the partial oxidation of propene [3].

The reactivity of large aggregates will usually be different from that of systems with a small extension in one or more dimensions like clusters or thin films [4, 5]. In such cases surface or interface atoms contribute significantly to the overall properties of the system, which has a strong effect on the electronic and geometric structure and thus onto the reactivity. Having the peculiarity of systems with a reduced dimensionality in mind, well-ordered monolayer thick MoO_3 islands have been prepared on Au(111) some years ago by Friend and co-workers [6–9]. Guided by computations, LEED (low-energy electron diffraction) and STM (scanning tunneling microscopy) results the authors concluded that the structure of the layers is similar to that of a single layer of α-MoO_3. Regular α-MoO_3 consists of pairs of such layers weakly interacting with the neighboring double layers via van der Waals forces.

In the context of studies of thin well-ordered α-MoO_3 layers on Au(111) [10] we prepared monolayer thick films which exhibit the same LEED pattern as the layers prepared by Friend et al [6–9]. These layers were studied with TDS (thermal desorption spectroscopy), XPS (X-ray photoelectron spectroscopy) [10], NEXAFS (near-edge X-ray absorption fine structure spectroscopy) and DFT (density functional theory) modeling of the NEXAFS spectra [11]. The I/V LEED (LEED spot intensity analysis) investigation discussed in this publication was started in

order to verify or improve the structural data of Friend et al [8], and thus to provide an improved structural input for DFT modeling. The I/V LEED structural optimization was performed with an evolutionary strategy algorithm, CMA-ES (covariance matrix adaption evolutionary strategy) [12], employing fully dynamical LEED I/V computations. This approach works well but is computationally expensive. CMA-ES is a population based robust local optimization method with a limited global optimization capability. In this context the question came up, how the performance of this method compares with that of other search methods used for I/V-LEED structure optimization methods. Therefore in the second part of this manuscript the performance of CMA-ES is evaluated and compared with that of two other population-based search methods: differential evolution (DE) [13] and a genetic algorithm (GA) as common global optimization methods, and with the local Powell optimization method based on tensor LEED computations.

Experimental

The measurements were performed in a chamber which contained a LEED system for I/V LEED measurements and sample characterization, Helmholtz coils for magnetic field compensation, a high-pressure cell for pressures of up to 1 bar, an electron-beam metal evaporator for the deposition of molybdenum, a sputter gun for sample preparation, a quadrupole mass spectrometer for residual gas analysis and thermal desorption spectroscopy, and a manipulator with a sample holder on which the sample was mounted. The LEED system was equipped with a channel plate electron multiplier combined with a phosphorous screen which permitted to record LEED images with small electron beam currents. Primary electron currents of some nA were sufficient to generate a reasonably intense LEED pattern. Low currents are important when sensitive materials like oxides are investigated since oxides may suffer from electron beam induced damage. A set of Helmholtz coils was mounted such that a homogeneous horizontal magnetic field could be

produced which was adjusted to compensate the earth magnetic field at the position of the sample. The magnetic field perpendicular to the path of the incident electrons must be very weak for I/V LEED measurements, since the incoming electrons have to impinge onto the sample along a direction which deviates from the surface normal by not more than a few tenth of a degree for all energies. Complete compensation of the magnetic field would have required an additional set of Helmholtz coils to also compensate the vertical earth field component. However, due to space constraints such coils could not be mounted. In order to minimize the effect of the non-compensated vertical magnetic field component, I/V LEED curves for high and low electron energies were recorded with slightly different sample alignment angles and matched in the overlapping energy range.

The need for perpendicular beam incidence required that the surface normal of the sample could be tilted, which was achieved by mounting the sample manipulator on a tilt mechanism, such that the whole manipulator could be tilted. The sample orientation was adjusted by minimizing the difference between I/V curves of symmetry-equivalent LEED spots.

The first step of the experimental determination of the I/V curves was the measurement of LEED patterns as a function of the electron energy with a step size of 1 eV, which was done with a commercially available software from OCI Vacuum Microengineering Inc. The I/V curves were derived from these images with a program which is able to perform spot-tracking and to integrate the intensity in a circular area around the spot maxima with sub-pixel accuracy. The I/V curves obtained this way were normalized to the beam current and subjected to a Sawatzky-Golay smooth of 2^{nd} or 4^{th} order depending on the noise of the data, followed by subtraction of a smooth background.

The Au(111) substrate was fixed on a molybdenum plate by two Mo sheets which were fitted into slits at the sides of the crystal. With these sheets the sample

was pressed towards the molybdenum plate to provide a good thermal contact and a well-defined position. The sample could be heated by electrons emitted from a glowing filament behind the plate if high voltage was applied to the plate or just by heat radiation from the filament if no high voltage was applied. Cooling with liquid nitrogen was possible by filling the hollow rod, on which the sample holder was mounted, with liquid nitrogen. A temperature of 110 K could be reached with this setup. At this temperature the I/V LEED curves were measured since the background intensity in LEED patterns is smaller at low temperature, which is especially true for gold due its small Debye temperature. For temperature measurement a K type thermocouple was attached to the Au(111) sample.

Standard sputtering/annealing sequences were employed to clean the Au(111) sample with the surface quality being judged from the quality of the LEED pattern. The MoO_3 layer was prepared according to a procedure described in reference 10. In the first step 0.5 MLE of molybdenum were deposited onto the Au(111) substrate (1 MLE $\triangleq$1 atom per gold surface atom) with a Mo flux of 0.2 MLE/min as calibrated with a quartz micro balance After this, the sample with the Mo layer was transferred into a high-pressure cell where the layer was oxidized for 10 minutes at 673 K in a stream of oxygen with a pressure of 50 mbar. Heating in the high-pressure cell was performed with a commercial halogen lamp equipped with a focusing gold mirror. During heating the sample was positioned into the focus of the mirror. After the end of the preparation procedure the lamp was switched off and the high-pressure cell was evacuated when the sample had reached a temperature of 373 K. Finally the sample was transferred back into the measuring chamber where a well-defined c(4×2) LEED pattern with sharp spots could be observed.

Computational

The LEED intensity computations were performed with the Barbieri/Van Hove Symmetrized Automated Tensor LEED (SATLEED) package [14]. For optimiza-

tion runs employing CMA-ES [12], differential evolution [13] and the genetic algorithm the SATLEED package was modified such that the calculated amplitudes of the diffracted beams were outputted and computations related to tensor LEED were skipped. Tensor LEED optimization runs were performed with the non-modified SATLEED package using the Powell optimization method as implemented in this package.

The agreement between the computed and the measured curves was quantified by the reliability factor (R-factor) defined by Pendry [15] (RP). Smaller R-factors mean better agreement with a value of zero indicating that the compared curves are identical. In IV-LEED studies R-factors below ~0.2 are usually considered to indicate that the model structure is similar to the real one. In the CMA-ES, the DE and the GA optimization runs the R-factor computation and the refinement of the structure by minimization of the R-factor were performed by a C++ program with the experimental data and the amplitudes of the diffracted beams computed by the modified SATLEED package as input. The program is able to subtract a linear or constant background from the experimental data, which was adjusted together with the other optimization parameters to minimize the R-factor. This was done in order to (partially) compensate for errors made in the initial background subtraction procedure. In the present case a linear background was subtracted. The required phase shifts were computed with the phase shift program accompanying the SATLEED package (see supplemental material, Figure SI5). The highest angular momentum value considered in the computations was $L_{max} = 9$. I/V curves were computed for an energy range from 60 to 400 eV with a 1 eV energy step width. For the calculation exploring the convergence ranges of the different methods L_{max} was set to 7, a constant background was optimized and the energy step width was set to 1.5 eV. In the case of the Powell method/tensor LEED optimization runs the R-factor routine implemented in the SATLEED package was used.

Finding the structure with the smallest difference between the computed I/V curves and the measured ones is essentially an optimization problem. In I/V LEED, an algorithm trying to find the structure under investigation will usually vary the coordinates of the atoms in a trial structure and possibly some other parameters like vibrational amplitudes and the inner potential in order to minimize the difference between the measured I/V curves and the I/V curves calculated for the trial structure. The parameters span a search space with a dimension equal to the number of parameters, and the volume equal to the product of the parameter variation ranges ($V = \prod_{i=1}^{D} \Omega_i$ with Ω_i being the variation range of parameter i and D the number of parameters).

The R-factor which maps a parameter vector to a real number ($\mathbb{R}^D \rightarrow \mathbb{R}$) defines a hyper surface in this space, and the optimization algorithm has to find the deepest minimum, the global minimum, of this surface. In general this hyper surface will not just have one minimum but many of them. Only one of them, the deepest one, is the desired global minimum and the other ones, the local minima, have to be disregarded by the search algorithm. The number of minima increases with the volume of the search space, such that a vast number of local minima may be expected for high-dimensional problems and large parameter variation ranges. A safe method to find the global minimum is a grid search on a sufficiently densely spaced grid in the search space. However, this is computationally too demanding in most cases, especially for high-dimensional problems, and therefore the search problem is usually made manageable by two measures:

1. *Use of an appropriate search method.*

   Search algorithms are usually designed such that minima on the hyper surface are attractive which means that the algorithm will try to adjust the parameters such that the R-factor approaches the minimum. This applies to local and global minima, and therefore the problem of identifying the global

minimum is not necessarily solved by this approach.

The multitude of such search methods may roughly be divided into local and global methods. If a local method starts the search in the convex area around a minimum, then it will just find this minimum and no other one. Therefore a good approximate knowledge of the location of the global minimum is required for the application of a local method. Global methods, on the other hand, promise to find the global minimum also from search starting points, which are not near to the global minimum. However, in real life it may take an unacceptably long time to find the global minimum with such a method from a search start point far away from the global minimum, and at least temporary trapping of the search in local minima may be expected. Therefore, even if global methods are applied it is relevant to start the search from a point not too far away from the global minimum. In general global search methods converge much slower than local ones.

2. *Restriction of the search space.*

   The search is confined to a sub-space of the whole parameter space in which the parameter set of the structure to be found is probably located. Such a sub-space is usually identified by chemical/physical intuition, possibly by assuming that the structure of the system under investigation is similar to that of another system with a known structure. A further reduction of the search space volume can be achieved by ignoring model structures with unphysical atomic distances. Also, proper consideration of the the symmetry is relevant since this usually reduces the dimension of the search space.

For the calculation of I/V curves, which are required for the computation of the R-factor, the tensor LEED approximation may suffice in many cases, but there are situations where R-factors calculated from tensor-LEED I/V curves exhibit significant errors even at a point not far away from the point for which the ref-

erence calculation was performed (see Figure SI1 in the supplemental material, especially curve c in the upper panel). To avoid this issue the tensor LEED approximation was not used for optimization runs employing global search methods. In these cases fully dynamical LEED calculation were performed for all points in the search space where the search algorithm required the computation of a R-factor. This is computationally much more demanding than tensor LEED, but parallel calculations on state-of-the-art compute clusters make this approach feasible. The tensor LEED approximation was only used for optimization runs employing the Powell method as implemented in the SATLEED package in order to get a clue on the performance of tensor LEED based local optimization in comparison with the other methods.

A large subset of the set of global optimization methods involves the optimization of a set of trial solutions. A large subset of these methods, the evolutionary strategies and the genetic algorithms, is inspired by the evolution of species in nature ("survival of the fittest"), which can be understood as an optimization process. As such, the set of trial structures is called population in these methods and the algorithms try to optimize the fitness of this population. Each individual is defined by its genome, which in the present case is simply a parameter set describing a trial structure for I/V LEED calculations, possibly plus some non-structural parameters as discussed later. Therefore, the population of individuals is essentially a population of trial structures. The individuals have a fitness which is calculated from the R-factor of the trial structures: the smaller the R-factor the better the fitness of the individual.

Three population-based search methods were employed: CMA-ES, DE and a real-valued GA. The GA is called real-valued because the genome consists of real numbers in the present case. For CMA-ES the Shark library [16] was used while the GA and DE codes were programmed by the authors. The I/V curve

computations for the individuals of the population were performed in parallel on a multi-processor system which reduced the computation time (but not the overall computational effort) significantly. Genetic algorithms have been employed before for LEED structural optimization [17–20], like differential evolution [21], and CMA-ES [22, 23] . The methods are derivative-free robust search methods, not very sensitive to discontinuities and noise.

Population-based search methods try to optimize the fitness of the population via production of offsprings, new individuals which may join the population only if they are able to pass a certain selection scheme which is usually based on the offsprings fitness. The genetic algorithm employs crossover and mutation operations for the production of offsprings combined with selection mechanisms which are designed to let individuals with a higher fitness survive with a higher probability on the one hand, and to preserve a certain genetic diversity on the other hand. The crossover operations combine the genomes of two individuals in certain ways to produce offsprings with new genomes and the mutation operations produce a new individual by modification of the genome of an individual of the population. Selection operations delete genetic material, crossover operations combine existing genetic material in a new way while mutations produce new genetic material. Thus, the genetic diversity within the population is defined by the balance of these operations. In the implementation used here, selection of the individuals to become parents and of the individuals to be replaced by offsprings was performed by a tournament selection scheme [24] preferring high fitness in the first case and low fitness in the second one.

The mutation and crossover operations of the GA are highly dependent on random numbers which control many aspects of these operations. In the present case the crossover operations ‘exchange of a random parameter between two individuals’, ‘exchange of parameters with 50% probability’, ‘one point crossover’

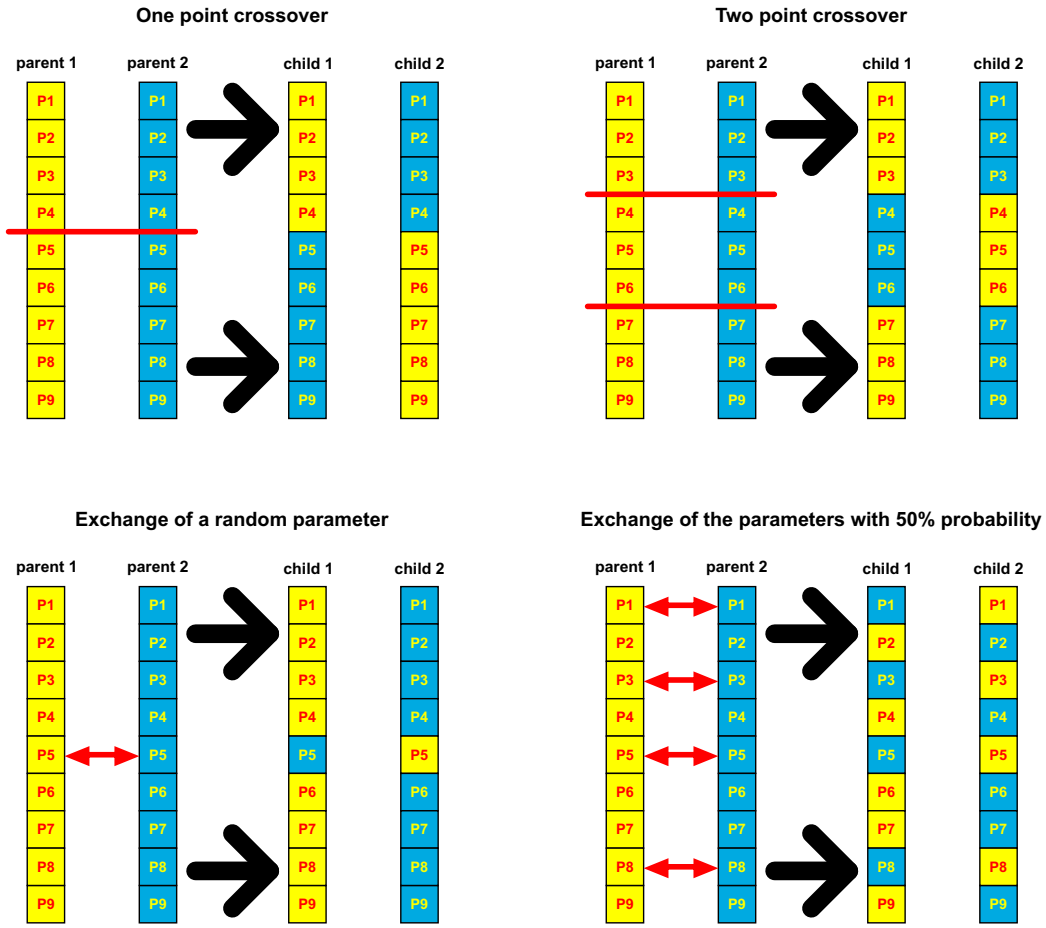

Figure 1: Schematic representation of the implemented crossover schemes for the genetic algorithm.

and 'two point crossover' as visualized in Figure 1 are implemented for the GA method. The implemented mutation operations for the GA are:

- Change of a random gene by a random value calculated using a Gaussian distribution.
- Change of a random gene by a random value calculated using an uniform distribution.
- Change of a randomly selected block of genes along a random direction in the search space with a distance calculated using a Gaussian random

distribution.

For the computation performed in the context of this I/V LEED study all listed crossover operations were executed with identical probabilities, likewise the mutation operations.

Differential evolution [13] is a rather simple but effective population based search scheme. In this method a set of next generation (G+1) mutants M_i^{G+1} is created from the individuals P_i^G of the population generation G according to

$$M_i^{G+1} = P_m^G + F \times (P_l^G - P_j^G) \tag{1}$$

with $i, m, l, j \in [0, NP]$ all different and m,l,j being integer random numbers. NP is the number of individuals in the population and i is a loop variable. $F \in [0, 2]$ is a pre-selected constant factor which controls the contribution of the difference $P_l^G - P_j^G$ to the mutants. In the next step a set of trial vectors T_i^{G+1} is computed according to

$$T_{ij}^{G+1} = \begin{cases} M_{ij}^{G+1} & \text{if RRN} < \text{CR or i} = \text{IRN} \\ P_{ij}^G & \text{otherwise} \end{cases} \tag{2}$$

$RRN \in [0, 1]$ is a real valued uniform random number which is drawn anew for every vector component, $IRN \in [1, D]$ is an integer valued random number which is drawn once per individual with D being the number of optimization parameters (=number of vector components). $j \in [1, D]$ is a loop index running over all vector components of the individuals. This procedure ensures that at least one parameter in each trial vector stems from a mutant. $CR \in [0, 1]$ is a pre-selected constant which controls the degree of inclusion of mutated genetic material into the population of trial vectors.

Finally the next generation of the population is produced by a simple selection scheme – an individual in the population is replaced by the corresponding trial

vector if the fitness of the individual is smaller than that of the trial vector and it is kept otherwise.

$$P_i^{G+1} = \begin{cases} M_i^{G+1} & \text{if Fitness}(M_i^{G+1}) > \text{Fitness}(P_i^G) \\ P_i^G & \text{otherwise} \end{cases} \tag{3}$$

The parameters CR and F control the performance of the algorithm – a large F will lead to mutants which differ significantly from the original individuals P_i^G while a small F produces mutants which are similar to them. Therefore, F controls the extension of the search space probed by the algorithm. A large F may lead to an improved global search capability, but may also lead to dilution of the population in the search space. The degree of acceptance of mutated genetic material, and thus the diversity of the population, increases with increasing CR, which therefore has a similar effect as an increasing F. The number of individuals also has an influence on the probability of finding the global minimum in that a larger population permits to sample the search space more thoroughly. For the differential evolution algorithm it is suggested that the number of individuals is between five and ten times the number of optimization parameters [13], which would have been between 120 and 240 in the present case. However, in order to keep comparability between the computations for the differential evolution algorithm and for the other methods, a population size of 20 was chosen, which, as will be shown later is large enough to make differential evolution a very competitive search algorithm.

In CMA-ES new individuals are produced exclusively by mutation. The genomes of the newly produced individuals are realizations of a multi-variate normal distribution which is defined by its covariance matrix. The parameters of the normal distribution, the covariance matrix, are adapted to the morphology of the R-factor hyper surface in the search space covered by the population. Due to this a higher convergence speed compared to that of the GA may be expected since

the latter method does not adapt its search strategy. The price of adaption may be a somewhat smaller chance to escape from local minima.

The parameters of the CMA-ES algorithm are the numbers of parents (μ) and offsprings (λ) with $\lambda > \mu$ as well as the initial standard deviations of the algorithm's normal distributions which, in the calculations leading to the results reported in the following, also define the distribution of individuals in the initial population. Such normal distributions were also used to set up the initial populations for GA and DE, and therefore their standard deviations were parameters which had to be chosen also in these cases. In GA the numbers of offsprings and parents had to be defined plus a number of other parameters like the widths and types (uniform, normal) of the distributions used for the computation of the random mutation step widths and parameters setting the selection pressure of the selection schemes. For DE only the population size and the parameters F and CR had to be supplied. The population size and the number of offsprings were defined such that not too many processor cores were needed for the optimization runs. 15-20 parallel computations per generation appeared to be reasonable. The other parameters were determined by test runs. However, GA has too many parameters for a thorough optimization. Therefore single parameter optimization test runs were performed for the most relevant parameters while the other parameters were set to some reasonable values. Elitism, which is the guaranteed survival of the individual(s) with the highest fitness was not applied in any search scheme since this leads to a higher chance that the search gets trapped in a local minimum for a prolonged time.

In order to avoid unnecessary computations, only structures with the symmetry suggested by the LEED pattern were accepted as individuals. Another implemented constraint was that only trial structures were considered, where the spacings between the atoms were larger than certain minimal distances. The minimum

distances were tabulated in a (3x3) matrix, such that different minimum spacings could be considered for different pairs of the three atom types (Au, Mo, O). The chosen minimal distances were inspired by known atomic and ionic radii.

Results and discussion

Structure

An image of the MoO_3 layer's LEED pattern taken with electrons with an energy of 45 eV is shown in Figure 2. The oxide produces a c(4×2) superstructure LEED pattern which is schematically illustrated in Figure 3. The black circle indicates the area of the experimental LEED pattern in Figure 2. Due to the symmetry of the Au(111) surface the oxide layer has three rotational domains which are shown in different colors in Figure 3. An important difference between the

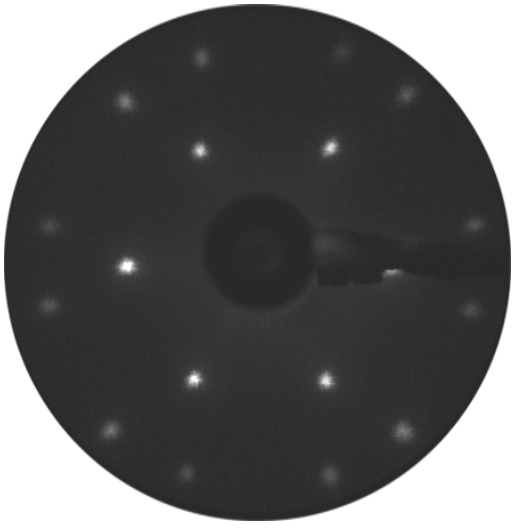

Figure 2: LEED pattern of c(4×2) MoO_3/Au(111). E_p=45 eV. The image was digitally enhanced by variation of contrast and brightness in order to make sure that all spots are clearly visible.

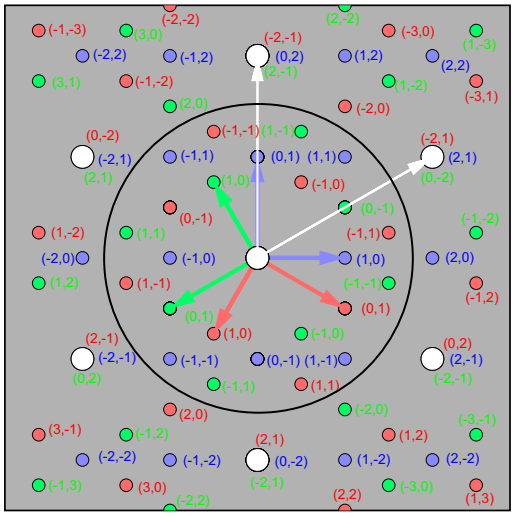

Figure 3: Simulated LEED pattern of c(4×2) MoO_3/Au(111). For the simulation the LEEDpat3 software [25] was used. Red, blue and green arrows and circles illustrate reciprocal unit vectors and LEED spots of different domains of the overlayer, while the Au(111) reciprocal unit vectors and LEED spots are drawn in white. The black circle marks the area of the LEED pattern shown in Figure 2.

scheme in Figure 3 and the LEED pattern in Figure 2 is that the (0,1) type spots are missing in the experimental LEED pattern. This indicates that the overlayer domains have a glide plane along the direction of the missing spots. As expected for a glide plane these spots do not show up at any electron energy. However, they get visible if the crystal is rotated by some degrees out of the glide plane which breaks the glide plane symmetry. The spots of the herringbone reconstruction of Au(111) [26] are not visible in the LEED pattern (Figure 2), which shows (1) that the reconstruction is lifted below the oxide layer due to the Au(111)-MoO_3 interaction and (2) that there are no large uncovered substrate areas.

Quek et al [8] have studied MoO_3 islands on Au(111) with STM and density functional theory. These islands exhibit a c(4×2) unit cell with missing (0,1) type spots [7] like the layers discussed here. Quek et al [8] argued that the structure of the islands might be similar to that of half of a double layer in α-MoO_3. The α-MoO_3 structure is illustrated in Figure 4A, and Figure 4B depicts the structure of a single layer (i.e. half of a double layer). Such a layer can be positioned onto

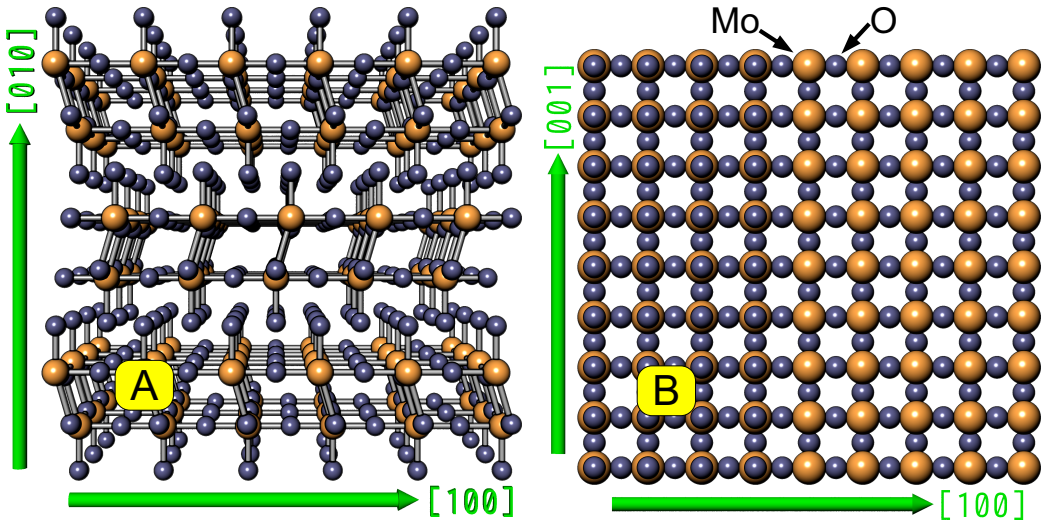

Figure 4: Geometric structure of α-MoO_3. (A): stacking of the double layers seen along [001]. (B): top view of a single layer (right half: without molybdenyl oxygen, left half: with molybdenyl oxygen).

Au(111) without much distortion such that a c(4×2) repeat unit results.

I/V LEED structural optimization computations with the CMA-ES optimization scheme were performed for a number of trial start structures consisting of MoO_3 single layers on Au(111) at different lateral positions on the substrate with the glide pane symmetry of the arrangement preserved in all cases. The optimization runs converged in most cases towards R-factors larger than 0.2, but for one position of the oxide layer on the substrate a R-factor of ~0.044 was obtained. This small R-factor is a good indication that the structure represented by this R-factor is the right one.

Figure 5 illustrates the structure of the system with the smallest R-factor. The Mo atoms are located on bridging positions on the Au(111) surface and the oxygen atoms are arranged similar to the positions they have in a single layer in α-MoO_3. This is somewhat different from the results obtained by Quek et al [8] who found

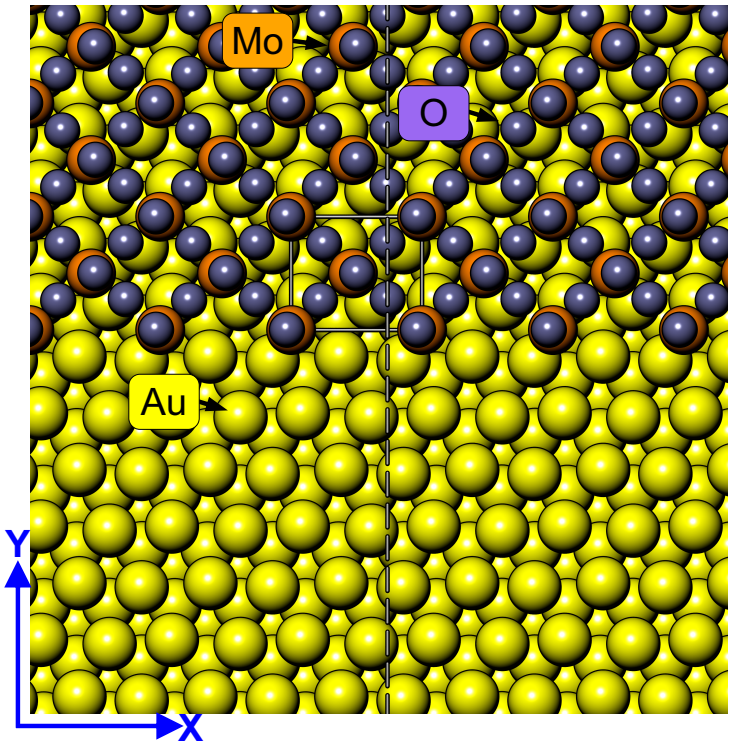

Figure 5: Graphical representation of the best-fit structure of a single layer of α-MoO_3 on Au(111). The overlayer unit cell (square) and a glide plane (dashed line) are indicated. In the bottom half of the image the optimized structure of the gold surface is shown without the overlayer. The directions of the x and y axes as used in this text are indicated.

Positions	X	Y	Z		X	Y	Z
$O_{molybdenyl}$	-1.34	1.31	4.24	Mo	-1.51	1.30	2.56
$O^{(1)}_{bridging}$	0.00	2.66	2.41	$O^{(2)}_{bridging}$	-2.78	2.60	2.38
$Au^{(1)}_{layer\,1}$	-0.71	-0.05	0.09	$Au^{(2)}_{layer\,1}$	-2.03	2.62	-0.02
$Au^{(1)}_{layer\,2}$	-0.74	1.67	-2.38	$Au^{(2)}_{layer\,2}$	-2.16	-0.84	-2.37
$Au^{(1)}_{layer\,3}$	-0.72	-1.66	-4.71	$Au^{(2)}_{layer\,3}$	-2.16	0.83	-4.71
$Au^{(1)}_{layer\,4}$	-0.72	0.00	-7.06	$Au^{(2)}_{layer\,4}$	-2.16	2.50	-7.06

Other parameters	Debye temperatures			Inner potential (imag)
	Mo	O	Au	
	421	382	137	4.74

Table 1: Parameters of the structure with the smallest Pendry R-factor. Debye temperatures are in Kelvin, the imaginary part of the inner potential is in eV and the coordinates are in Å. The table at the top lists only data for symmetry-inequivalent atoms. The coordinates of the bottom four atoms ($Au_{layer\,3}$ and $Au_{layer\,4}$) were set to the bulk positions and kept fixed in the optimization run. They are listed here to provide a reference for the coordinates of the atoms in the layers above.

that the Mo atoms are located in threefold hollow sites on the Au(111) surface. In that case the layer had a quasi glide plane rotated by 90° with respect to the one shown in Figure 5. In the present case the overlayer glide plane matches a substrate glide plane. The system does not have any mirror planes, and therefore the symmetry of the overlayer is pg. The gold atoms in the two top layers which were included in the optimization were found to be slightly displaced by distances in the range of a tenth of an Ångstrøm.

For the LEED computations seven experimental I/V curves with a total energy range of 1616 eV were considered. Each of these curves is the sum of the intensities of symmetry-equivalent spots which were summed up for the three different domains of the layer. The experimental curves are shown in Figure 6 together with the computed curves for the structure with the smallest R-factor. The parameters for the structural optimization process were the atomic coordinates of all atoms of the overlayer and the first two gold layers (2 molybdenum atoms, 6 oxygen atoms, and 8 Au atoms per c(4×2) unit cell). The presence of the glide plane reduces the

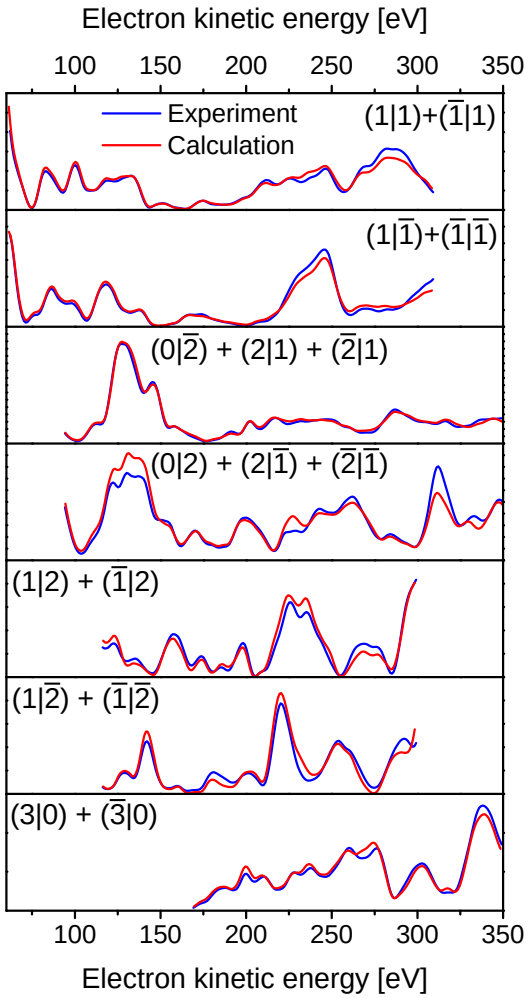

Figure 6: Experimental data and calculated I/V curves for the structure with the smallest R-Factor.

numbers of independent atomic positions by a factor of two such that 24 structural parameters (8 atoms, each with 3 coordinate values) had to be optimized. Further optimization parameters were the imaginary part of the inner potential and the Debye temperatures of the three elements involved (Mo, O, Au), giving a total number of 28 optimization parameters. The best-fit parameters are listed in Table 1.

Convergence speed of CMA-ES, differential evolution and the genetic algorithm

This chapter compares the convergence speeds of CMA-ES, DE and GA for the case of the MoO_3 layer on Au(111) discussed in this text. In the case of CMA-

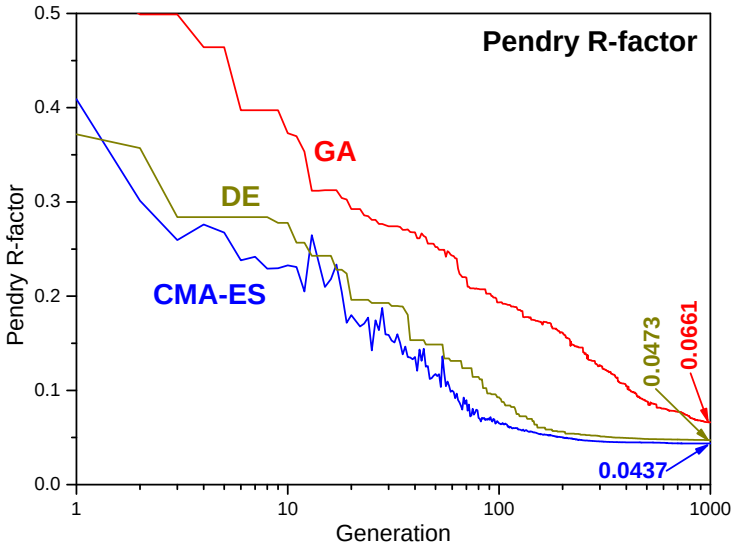

Figure 7: Pendry R-factor of the fittest individual as a function of the number of generations for CMA-ES, differential evolution and the genetic algorithm.

ES the number of individuals in the population and the number of offsprings were set to 8 and 17, respectively, as suggested for this number of parameters. In every next generation the 8 best offsprings replace the parents and become the new population. Thus, for every generation 17 I/V curves had to be computed. In the case of the genetic algorithm a population size of 30 was chosen. In each generation 12 offsprings were produced by crossover and 3 by mutations. These were mixed into the new generation, replacing 15 parents. In this case 15 computations of I/V curves had to be performed in every generation. The population size in the case of the DE algorithm was set to 20, which is also the number of I/V curve calculations performed in every generation. Optimization of the imaginary part of the inner potential and of the Debye temperatures was not implemented in the case of the GA program and therefore these parameters were set to the numbers given in Table 1. In order to establish comparability between the methods this was also done in the CMA-ES and differential evolution runs.

The Pendry R-factor of the fittest individual is plotted as a function of the number of generations in Figure 7 for computations employing the CMA-ES method, differential evolution and the genetic algorithm (The R-factor of the fittest individ-

uals as a function of the number of fully dynamical computation is shown in the supplemental material, Figure SI2). The convergence speeds of the three methods are different with the genetic algorithm being slowest and the two other methods having somewhat similar performance with CMA-ES being slightly faster. CMA-ES reaches a R-factor of less than 0.1 after 60 generations, differential evolution requires 129 generations while the genetic algorithm needs 417 generations. Especially, the genetic algorithm is very slow at high generation numbers where the structural fine tuning takes place. Here complex small parameter changes are required to approach the global minimum. This can be done much better by CMA-ES which adapts the search strategy to the local topography of the R-factor hyper surface and by DE due to the contraction of the population in the search space near to the global minimum and the consequent reduction of the mutation distances. Thus we conclude that search strategy adaption is relevant especially for the final steps of the search. We note that mutation width adaption according to a "20% of the mutated offsprings must be fitter than their parents" rule as sometimes employed in evolutionary strategy computations is able to increase the GA convergence speed significantly, though it does not reach the performance of the two other methods (not shown here).

A parameter which is relevant for the convergence speed and the ability of a search method to find the global minimum is the integration of some kind of randomness into the search algorithm. All three population based search methods discussed here make heavy use of random number generators to produce new generations, which introduces randomness into the populations. This permits to explore remote search space areas, probably not explored without randomness. Individuals somewhat remote from the center of the population can also help the search to escape from local minima. However, these advantages have a price, a reduced convergence speed due to the presence of these remote individuals. The

randomness of CMA-ES and DE adapts to the search space such that it is smaller near to a minimum, i.e. the average mutation distances decrease. The impact of randomness on the convergence speed is smaller for these methods than for GA where the mutation step width distributions, and thus the randomness introduced by mutations, are constant, independent of the local search space and the distribution of individuals in the population, which is another reason for the observed lower convergence speed of GA.

Accuracy

Different CMA-ES optimization runs (full parameter sets including Debye temperatures and the imaginary part of the inner potential) which were stopped after having reached R-factors below 0.05 gave Debye temperatures different by up to 30 K. The corresponding spread of the imaginary part of the inner potential was in the range of 0.3 eV. These numbers may give an impression of the accuracy of the obtained Debye temperatures and the imaginary part of the inner potential.

The spread of the coordinate values among the three methods after 1000 iterations (R-factors=0.0437, 0.0449 and 0.0661, see Figure 7) is $\Delta x = 8.4 \times 10^{-3}$ Å, $\Delta y = 5.7 \times 10^{-2}$ Å, $\Delta z = 5.5 \times 10^{-4}$ Å. For the other atoms the results are similar regarding size and order of the differences (x medium, y largest, z smallest). These differences shed some light onto the accuracy of the coordinates. The small difference of the z coordinate values is indicative of a good accuracy, which may be traced back to the high sensitivity of the IV curves to the z coordinates of the atoms due to the geometry of the experiment with the primary beam traveling along the z axis and the diffracted beams detected within only ±35° around the z-axis. The accuracy of the horizontal coordinates is much smaller with the accuracy of the y coordinate being smaller than that of the x-coordinate which may be due to glide plane related absence of the (0,1) type spots. These could therefore not be considered in the computations which reduces the weight of the

y coordinates in the R-factor computation.

Convergence ranges of CMA-ES, the genetic algorithm, differential evolution and tensor LEED

A very relevant parameter of a search method is the probability of finding the global minimum as a function of the distance between the global minimum and the position from which the search is started. A reasonable probability also for somewhat larger distances helps to find the global minimum even if the first guess of the structure is not very near to the real structure. In order to get an idea of the probability of convergence towards the global R-factor minimum as a function of the distance between the search starting point and the global minimum, optimization runs were performed with start configurations randomly distributed in sub-spaces with parameters $P_i \in [B_i - \Delta_i/2 \ldots B_i + \Delta_i/2]$ centered around the best-fit atomic configuration with parameters B_i. For $\Delta_i = \Delta$, i.e. all Δ_i set to the same value this leads to a distribution of euclidic distances between the start structures and the global minimum with a well-defined probability maximum at $\approx 1.4\Delta$ and a half-width of $\approx 0.3\Delta$ (see supplemental material, Figure SI3). The parameters of the start population's individuals were centered around these start positions following normal distributions with standard deviations σ_i. Like the Δ_i, the σ_i were all set to the same value, $\sigma_i = \sigma$. In short, Δ defines the average distance between the start population and the global minimum and σ defines the spread of the individuals in the initial population. In all runs, the Debye temperatures and the imaginary part of the inner potential were set to the best-fit values listed in Table 1.

The optimal parameters F and CR for the differential evolution algorithm were determined in a series of test runs in which F and CR as well as σ were varied for fixed Δ=0.8 Å. This means that the mean difference between the best fit coordinate values and the corresponding values in the start population individuals is 0.4 Å, which is a distance somewhat beyond the distance which tensor-LEED can

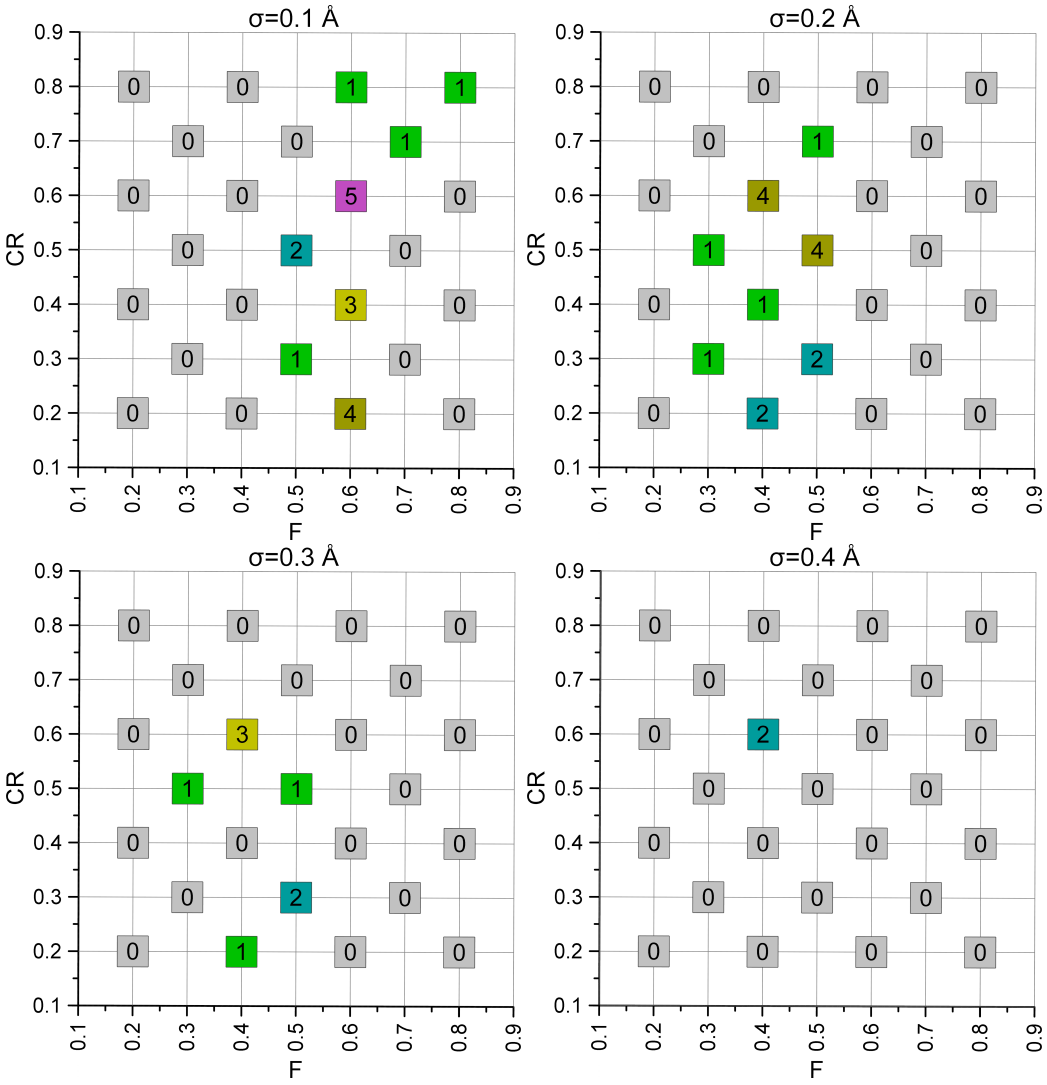

Figure 8: Number of converged runs (RP<0.1) out of seven differential evolution runs for different values of the parameters F, CR and σ. The parameter Δ was set to 0.8 Å in all runs.

commonly handle. Figure 8 shows that the success rates are largest for σ=0.1 and 0.2 Å with F and CR both being about 0.5-0.6. The DE computations discussed in the following were performed with F=CR=0.5.

Powell method/tensor LEED optimization was performed for different Δ's with the non-modified SATLEED package [14]. The runs were iterated by using the structure obtained in an optimization run as start structure for the following

CMA-ES	σ = 0.05 Å						σ = 0.1 Å					
Δ [Å]	RP<0.1		RP<0.15		RP<0.2		RP<0.1		RP<0.15		RP<0.2	
0.2	7	7	7	7	7	7	7	7	7	7	7	7
0.3	5	5	5	5	6	6	7	7	7	7	7	7
0.4	6	6	6	6	6	6	7	7	7	7	7	7
0.5	4	4	4	4	4	4	4	4	4	4	5	5
0.6	1	1	1	1	1	1	3	3	3	3	3	3
0.7	3	3	3	3	3	3	2	2	2	2	3	3
0.8	0	0	0	0	0	0	0	0	0	0	1	1
Σ	26	26	26	26	27	27	30	30	30	30	33	33

CMA-ES	σ = 0.2 Å						σ = 0.3 Å					
Δ [Å]	RP<0.1		RP<0.15		RP<0.2		RP<0.1		RP<0.15		RP<0.2	
0.2	3	3	3	3	3	3	1	1	1	1	1	1
0.3	0	0	0	0	0	0	0	0	0	0	0	0
0.4	1	1	1	1	1	1	0	0	0	0	0	0
0.5	0	0	0	0	2	2	0	0	0	0	0	0
0.6	1	1	1	1	1	1	0	0	0	0	0	0
0.7	0	0	0	0	1	1	0	0	0	0	0	0
0.8	0	0	1	1	1	1	0	0	0	0	0	0
Σ	5	5	6	6	9	9	1	1	1	1	1	1

Table 2: Convergence properties of the CMA-ES algorithm (without optimization of Debye temperatures and the imaginary part of the inner potential). Each double column list the number of runs out of seven runs which have reached a R-factor smaller than the one given in the second row after 300 iterations (first column) and 600 iterations (second column). For details see text.

run until the difference of the R-factors obtained in subsequent runs was smaller than 0.001.

Tables 2 (CMA-ES), 3 (DE, F=CR=0.5), and 4 (GA) list for a number of (Δ, σ) combinations the number of runs out of seven runs which have reached Pendry R-factors of less than 0.1, 0.15, and 0.2 after 300 and 600 generations. Tables for some other F and CR combinations are shown in the supplemental material, Tables SI1-SI3. We note that there is an obvious statistical noise in the numbers given in the tables which is related to the limited number of optimization runs per parameter combination. This limitation was imposed by the finite availability of

DE	σ = 0.05 Å						σ = 0.1 Å					
Δ [Å]	RP<0.1		RP<0.15		RP<0.2		RP<0.1		RP<0.15		RP<0.2	
0.2	7	7	7	7	7	7	7	7	7	7	7	7
0.3	7	7	7	7	7	7	7	7	7	7	7	7
0.4	6	6	6	6	6	6	6	6	6	6	6	6
0.5	5	5	5	5	5	5	7	7	7	7	7	7
0.6	2	2	2	3	3	3	3	3	3	3	3	3
0.7	3	3	3	3	3	3	1	1	1	1	1	1
0.8	0	0	0	0	0	0	2	2	2	2	4	4
Σ	30	30	30	31	31	31	33	33	33	33	35	35

DE	σ = 0.2 Å						σ = 0.3 Å					
Δ [Å]	RP<0.1		RP<0.15		RP<0.2		RP<0.1		RP<0.15		RP<0.2	
0.2	7	7	7	7	7	7	1	3	1	3	1	3
0.3	7	7	7	7	7	7	0	2	0	2	1	2
0.4	4	6	5	6	5	6	0	2	0	2	0	2
0.5	2	7	4	7	5	7	0	0	0	0	0	0
0.6	4	5	4	5	4	5	0	1	0	1	0	1
0.7	1	3	1	4	1	4	0	0	0	0	0	0
0.8	0	0	0	0	0	0	0	0	0	0	0	0
Σ	25	35	28	36	29	36	1	8	1	8	2	8

Table 3: Convergence properties of the differential evolution algorithm (without optimization of Debye temperatures and the imaginary part of the inner potential). The parameters F and CR were both set to 0.5. Each double column list the number of runs out of seven runs which have reached a R-factor smaller than the one given in the second row after 300 iterations (first column) and 600 iterations (second column). For details see text.

computational resources.

The σ's and the Δ's clearly have a significant effect onto the probability of convergence. For the Δ parameter the dependence is such that the success rate simply decreases with increasing Δ. For the σ parameter, which defines the width of the initial population, it appears that the probability of convergence for medium range Δ is somewhat enhanced if σ is not too small. This kind of $\Delta - \sigma$ correlation is expected since for a given Δ there is a reasonable chance that the start population includes individuals with at least some parameters in the convex area near to their

GA	σ = 0.05 Å						σ = 0.1 Å					
Δ [Å]	RP<0.1		RP<0.15		RP<0.2		RP<0.1		RP<0.15		RP<0.2	
0.2	7	7	7	7	7	7	7	7	7	7	7	7
0.3	7	7	7	7	7	7	7	7	7	7	7	7
0.4	3	6	5	6	6	6	7	7	7	7	7	7
0.5	0	1	0	1	0	1	3	6	5	6	5	6
0.6	0	0	0	0	0	0	1	2	2	2	2	3
0.7	0	0	0	0	0	0	0	2	1	2	2	2
0.8	0	0	0	0	0	0	0	0	0	0	0	0
Σ	17	21	19	21	20	21	25	31	29	31	30	32

GA	σ = 0.2 Å						σ = 0.3 Å					
Δ [Å]	RP<0.1		RP<0.15		RP<0.2		RP<0.1		RP<0.15		RP<0.2	
0.2	6	7	6	7	6	7	0	1	1	3	1	3
0.3	5	6	6	6	7	7	0	4	2	4	2	4
0.4	4	5	4	5	5	5	0	2	1	2	1	2
0.5	1	2	1	2	1	2	1	1	1	1	1	1
0.6	2	2	2	2	2	3	0	0	0	0	0	0
0.7	0	1	1	1	1	1	0	0	0	0	0	0
0.8	0	0	0	0	0	0	0	0	0	0	0	0
Σ	18	23	20	23	22	25	1	8	5	10	5	10

Table 4: Convergence properties of the genetic algorithm. Each double column list the number of runs out of seven runs which have reached a R-factor smaller than the one given in the second row after 300 iterations (first column) and 600 iterations (second column). For details see text.

Tensor LEED			
Δ [Å]	RP<0.1	RP<0.15	RP<0.2
0.2	7	7	7
0.3	6.7	6.7	6.7
0.4	4.7	5	5
0.5	2.7	2.7	2.7
0.6	0.3	0.3	0.3
0.7	0	0	0
0.8	0	0	0
Σ	21.3	21.7	21.7

Table 5: Convergence properties of Powell method/tensor LEED. For details see text.

global minimum values only if σ is large enough.

The difference between the success rates after 300 and 600 generations is an indicator for the convergence speed. In agreement with Figure 7, CMA-ES and DE are apparently faster than GA. The similarity of the numbers in the RP<0.1, RP<0.15 and RP<0.2 columns demonstrates that in most cases the search algorithms directly approach the global minimum if they could reach a point in the search space with RP<0.2, which is with a high probability a point in the convex area around the global minimum.

The winner of this contest with respect to the convergence range is clearly the differential evolution algorithm which could find the global minimum for Δ=0.8 Å in a significant number of cases. Therefore one would likely resort to this method if the knowledge about the structure to be found is rather limited. In the case of CMA-ES only a few runs succeeded for Δ=0.8 Å, while GA could not produce any success for this Δ value. To repeat the role of Δ: this is the parameter which defines the mean euclidic distance between the center of the start population and the global minimum. However, if it comes to converge speed, especially in the region near to the global minimum, then CMA-ES is the winner among the global methods, which is probably due to the more sophisticated search strategy tuning of this method as compared to the other ones. GA lags somewhat behind which is a result also obtained by other authors for different optimization problems (see for instance [27]).

Table 5 lists how many runs out of 21 tensor LEED runs combined with the Powell search method stopped with R-factors below 0.1, 0.15, and 0.2 for different Δ's. To simplify comparison with the results for the other algorithms, where only seven runs were performed due to the higher computational effort, the numbers in Table 5 were divided by three. The success rate lags clearly behind the success rates of the other methods for $\sigma = 0.1$ (see the Σ columns in Tables 2, 3, and 4),

dropping to a very small value for Δ=0.6 Å and vanishing above. This is not a small difference to DE since in a 24-dimensional search space the ratio of the subspace volumes in which success may be expected with a reasonable probability is the ratio of the Δ's (0.8 and 0.5 Å for DE and tensor leed/Powell, respectively) to the power of 24, which is about 80000 in the present case. On the other hand, the computational effort for tensor LEED is much smaller which may compensate for this in many cases, especially if a compute cluster for parallel computations is not available. Therefore, tensor LEED/Powell method (or any other local search method) based structural optimization is attractive in case that the structure from which the search is started is expected to be not much different from the global minimum structure.

The DE and CMA-ES implementations used in this study can optimize the Debye temperatures and the imaginary part of the inner potential in addition to the structural parameters, but it is surely possible to implement optimization of such non-structural parameter also for the iterated Powell method/tensor LEED search method. Above $\Delta \approx$ 0.4 Å, the convergence rate decreases significantly for all methods since the convex area around the minimum has a width of $\approx$ 0.4 Å for the z coordinates (they are wider for the x and y coordinates, see supplemental material, Figure SI4) which means that for $\Delta \geq$ 0.4 Å there is an increasing chance that some parameters of the individuals refer to positions outside of the convex area around the global minimum. For the local Powell method it is essentially impossible to find the global minimum from such a starting point, and apparently the probability of failure increases also for the global methods. Using larger populations with a larger spread of the individuals will probably help, but this will also increase the computational effort. Yet untested is the use of tensor-LEED in combination with a global search method with an appropriately set limit of the difference between the structure for which the reference calculaton is per-

formed and the structure for which the tensor LEED computation is performed (in order to limit the error related to the tensor LEED approximation). Due to the limited computational effort of tensor LEED one may expect that this approach is competitive.

Summary

We have investigated the structure of a c(4×2) MoO_3 monolayer on Au(111) with I/V LEED structural analysis employing the CMA-ES evolutionary strategy. The structure of the c(4×2) MoO_3 layer is very similar to that of the single layers constituting the double layers found in α-MoO_3. A glide plane in the structure leads to spot extinctions in the LEED pattern.

The experimental I/V-LEED results were used to comparatively evaluate the convergence speed and the convergence ranges of CMA-ES, a genetic algorithm, differential evolution and tensor LEED combined with the Powell optimization scheme. The latter is, in contrast to the three other methods, a local optimization method. The best choice with respect to the convergence range is the differential evolution scheme. Regarding convergence speed CMA-ES is superior while the genetic algorithm lags behind both methods in both respects. The convergence range of tensor LEED is somewhat smaller than that of the global methods while the computational effort is much smaller. Therefore in a number of cases the use of tensor LEED for the determination of a structure with I/V-LEED might still be preferable, but in cases where the initial structural guess is expected to be connected with a significant uncertainty global methods become relevant. The differential evolution and CMA-ES implementation used in this study are enabled to optimize arbitrary non-structural parameters like Debye temperatures and the imaginary part of the inner potential which is not possible with the employed tensor LEED implementation, but there are no general issues preventing the optimization of such non-structural parameters also in combination with tensor-LEED.

Acknowledgments

We acknowledge the DFG for funding through their collaborative research center 546, “Transition Metal Oxide Aggregates”.

References

[1] A. P. V. Soares, M. F. Portela, A. Kiennemann, Methanol selective oxidation to formaldehyde over Iron-Molybdate catalysts, Catal. Reviews 47 (2004) 127.

[2] M. Bowker, A. F. Carley, M. House, Contrasting the behaviour of MoO_3 and MoO_2 for the oxidation of methanol, Surf. Sci. 120 (2008) 34.

[3] T. Ressler, J. Wienold, R. E. Jentoft, F. Girgsdies, Evolution of defects in the bulk structure of MoO_3 during the catalytic oxidation of propene, Eur. J. Inorg. Chem. 2003 (2003) 301.

[4] C. B. Roldan, Synthesis and catalytic properties of metal nanoparticles: Size, shape, support, composition, and oxidation state effects, Thin Solid Films 518 (12) (2010) 3127–3150.

[5] F. Vines, J. R. B. Gomes, F. Illas, Understanding the reactivity of metallic nanoparticles: beyond the extended surface model for catalysis, Chem. Soc. Rev. 43 (2014) 4922–4939.

[6] M. M. Biener, C. M. Friend, Heteroepitaxial growth of novel MoO_3 nanostructures on Au(111), Surf. Sci. 559 (2004) L173.

[7] M. M. Biener, J. Biener, R. Schalek, C. M. Friend, Growth of nanocrystalline MoO_3 on Au(111) studied by in situ scanning tunneling microscopy, J. Chem. Phys. 121 (2004) 12010.

[8] S. Y. Quek, M. M. Biener, J. Biener, C. M. Friend, E. Kaxiras, Tuning electronic properties of novel metal oxide nanocrystals using interface interactions: MoO_3 monolayers on Au(111), Surf. Sci. 577 (2005) L71.

[9] X. Deng, S. Y. Quek, M. M. Biener, J. Biener, D. H. Kang, R. Schalek, E. Kaxiras, C. M. Friend, Selective thermal reduction of single-layer MoO_3 nanostructures on Au(111), Surf. Sci. 602 (2008) 1166.

[10] S. Guimond, D. Göbke, J. M. Sturm, Y. Romanyshyn, H. Kuhlenbeck, M. Cavalleri, H.-J. Freund, Well-ordered molybdenum oxide layers on Au(111): Preparation and properties, J. Phys. Chem. C 117 (2013) 8746–8757.

[11] M. Cavalleri, K. Hermann, S. Guimond, Y. Romanyshyn, H. Kuhlenbeck, H.-J. Freund, X-ray spectroscopic fingerprints of reactive oxygen sites at the MoO_3(010) surface, Catal. Today 124 (2007) 21.

[12] N. Hansen, The CMA evolution strategy: a comparing review, in: J. A. Lozano, P. Larranaga, I. Inza, E. Bengoetxea (Eds.), Towards a new evolutionary computation. Advances on estimation of distribution algorithms, Springer, 2006, pp. 75–102.

[13] R. Storn, K. Price, Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces, Journal of Global Optimization 11 (1997) 341–359.

[14] A. Barbieri, M. A. Van Hove, Symmetrized Automated Tensor LEED Package (SATLEED), downloaded from http://www.icts.hkbu.edu.hk/vanhove/.

[15] J. B. Pendry, Reliability factors for LEED calculations, J. Phys. C: Solid St. Phys. 13 (1980) 937.

[16] C. Igel, V. Heidrich-Meisner, T. Glasmachers, Shark, Journal of Machine Learning Research 9 (2008) 993–996.

[17] E. A. Soares, C. M. C. de Castilho, V. E. de Carvalho, Advances on surface structural determination by LEED, J. Phys.: Condens. Matter 23 (2011) 303001.

[18] R. Döll, M. A. VanHove, Global optimization in LEED structure determination using genetic algorithms, Surf. Sci. 355 (1996) L393–L398.

[19] M. L. Viana, W. Simões e Silva, E. A. Soares, V. E. de Carvalho, C. M. C. de Castilho, M. A. Van Hove, Scaling behavior of genetic algorithms applied to surface structural determination by LEED, Surf. Sci. 602 (2008) 3395–3402.

[20] M. L. Viana, D. D. dos Reis, E. A. Soares, M. A. Van Hove, W. Moritz, V. E. de Carvalho, Novel genetic algorithm search procedure for LEED surface structure determination, J. Phys.: Cond. Mat. 26 (2014) 225005.

[21] V. B. Nascimento, E. W. Plummer, Differential evolution: Global search problem in LEED-IV surface structural analysis, Materials Characterization 100 (2015) 143 – 151.

[22] F. E. Feiten, J. Seifert, J. Paier, H. Kuhlenbeck, H. Winter, J. Sauer, H.-J. Freund, Surface structure of V2O3(0001) revisited, Phys. Rev. Lett. 114 (2015) 216101.

[23] F. E. Feiten, H. Kuhlenbeck, H.-J. Freund, Surface structure of V2O3(0001): A combined i/v-leed and stm study, J. Phys. Chem. C 119 (2015) 22961–22969.

[24] J. Yang, C. K. Soh, Structural optimization by genetic algorithms with tournament selection., J. Comp. Civil Eng. 11 (1997) 195.

[25] K. Hermann, M. A. Van Hove, LEEDPat3 software, 2010.

[26] B. K. Min, X. Deng, D. Pinnaduwage, R. Schalek, C. M. Friend, Oxygen-induced restructuring with release of gold atoms from Au(111), Phys. Rev. B 72 (2005) 121410.

[27] C. Cai, H. Jiang, Performance comparisons of evolutionary algorithms for walking gait optimization, in: 2013 International Conference on Information Science and Cloud Computing Companion (ISCC-C), 2013, pp. 129–134.