Using Pattern Search Methods for Surface Structure Determination of Nanomaterials

Zhengji Zhao
High Performance Computing Research Department
Lawrence Berkeley National Laboratory
1 Cyclotron Road Mail Stop 50F1650
Berkeley, CA 94720, USA
(email: *ZZhao@lbl.gov*).

Juan C. Meza
High Performance Computing Research Department
Lawrence Berkeley National Laboratory
1 Cyclotron Road Mail Stop 50F1650
Berkeley, CA 94720, USA

M. V. Hove
Department of Physics and Materials Science
City University of Hong Kong, Hong Kong

Abstract

Atomic scale surface structure plays an important role in describing many properties of materials, especially in the case of nanomaterials. One of the most effective techniques for surface structure determination is low-energy electron diffraction (LEED), which can be used in conjunction with optimization to fit simulated LEED intensities to experimental data. This optimization problem has a number of characteristics that make it challenging: it has many local minima, the optimization variables can be either continuous or categorical, the objective function can be discontinuous, there are no exact analytic derivatives (and no derivatives at all for categorical variables), and function evaluations are expensive. In this study, we show how to apply a particular class of optimization methods known as pattern search methods to address these challenges. These methods do not explicitly use derivatives, and are particularly appropriate when categorical variables are present, an important feature that has not been addressed in previous LEED studies. We have found that pattern search methods can produce excellent results, compared to previously used methods, both in terms of performance and locating optimal results.

1 Introduction

The surface structure determination of nanostructures is important because many of their properties (e.g., mechanical, electrical, magnetic, chemical and optical) depend to some extent on the atomic-scale structure of the surface. For example, a surface formed by adding lithium to the surface of a nickel crystal can be used as a catalyst. The key to understanding and controlling the catalytic activity however is to have exact knowledge about how these two types of atoms (Li and Ni) occupy the surface. By surface structure here, we mean the geometric positions of the atoms within the several atomic layers (or a nanometer-scale distance) from the surface and the chemical identities of these atoms. As a result of the surface reconstruction, these atoms could occupy any geometric positions within several atomic layers from the surface.

Among the various methods [1, 2] of determining the surface structure, one of the principal techniques is the low energy electron diffraction (LEED) method [3, 4, 5] which combines experimental and theoretical analysis. This is mainly due to the fact that the LEED experiment is inexpensive compared to most other techniques, while still providing high precision for the determination of the full surface structure. In fact, some methods are only able to determine a subset of the surface structure. In a LEED experiment, an electron beam is emitted from an electron gun (usually with an energy of 20eV–600eV) and hits the surface of the test sample. By recording the reflected beams (see Fig. 1), one can obtain the electron diffraction pattern, which includes enough information to be used to accurately determine the surface structure. Given a trial surface structure, one can also compute a LEED pattern by calculating the multiple scattering of the incoming electron beam due to these surface atoms. By comparing the simulated LEED pattern with the experimental data, one can iterate until a surface structure is determined that matches the experimental diffraction patterns.

The LEED method can therefore be formulated as an inverse optimization problem subject to some constraints. In this problem, the optimization problem contains two types of parameters: continuous and categorical and the objective function to be minimized is the misfit between the calculation and the experiment. The continuous variables correspond to the atomic positions and the categorical variables correspond to the chemical identities of the atoms. By a categorical variable we mean one that must be chosen from among two or more categories, but where there is no intrinsic ordering to the categories. The major distinction between these and discrete variables is that the lack of an intrinsic ordering precludes us from using standard mixed integer programming methods. These types of optimization problems are known as mixed variable problems because of the two types of parameters that need to be considered. The inverse problem has a number of characteristics that make it challenging however: the objective function is not smooth (due to the presence of the categorical variables and invalid points in the parameter space, and also due to the numerical effects on a fine scale), there exist many local minima, derivative information is difficult to get or unavailable, and the function evaluations are expensive.

A number of approaches to automated searches for the globally best solution have been explored and applied over the years using LEED, including: simulated annealing [6], fast simulated annealing [7, 8, 9, 10], and a modified random sampling algorithm [11]. These methods, which allow global optimization, were, however, limited to relatively simple problems in quite restricted search domains. In particular, no element identification is normally attempted, avoiding the fitting of categorical variables. As implemented, these methods tend to be very compute-intensive, requiring the evaluation of large numbers of trial solutions.

The most relevant previous attempts [12, 13] to solve this mixed variable optimization prob-

lem used genetic algorithms (GAs) which simulate the natural evolution of living organisms. In this algorithm, each candidate solution, which is denoted by a chromosome, is assigned a "fitness" value. Chromosomes with a lower function value are assigned a higher fitness value. The algorithm starts with a population of random chromosomes and applies a selection rule, which combines chance and a preference for the chromosomes with high fitness values, to choose pairs of chromosomes to be used as parents for the next generation. The parents are then used to produce the next generation of chromosomes (children) following certain crossover and mutation rules. The hope is that, following the analogy of survival of the fittest, the better parents will produce better children. The population is then updated by adding the children to the population. There are many different ways to update the population; children may simply replace the parents or allow parents to compete with their children so that only those chromosomes with higher fitness values are kept in the population. This process can be repeated until a chromosome with a satisfactorally high fitness value is found. For more details about GAs, we refer to [14, 15, 16].

Döll and Van Hove first applied GAs to the determination of a Ir(110)-(1x2) missing row surface structure [12]. Since the test problem had a rather small parameter space ($1\mathring{A} \times 1\mathring{A} \times 1\mathring{A}$), they were able to perform an exhaustive grid search (on a grid of $0.05\mathring{A}$ fineness). They found that GAs could locate the global minimum 10 times faster than an exhaustive grid search, and performed better than simulated annealing [6]. They therefore concluded that GAs could be used as a global optimizer for the LEED surface structure determination [12]. Later, Stone applied GAs to the complex surface Ni(001)-(5x5)-Li [17]. This surface is formed by adding lithium atoms to a nickel crystal surface. Because the parameter space volume is around $12.5\mathring{A} \times 12.5\mathring{A} \times 5\mathring{A}$, an exhaustive search is not viable. However using a parallel GA package that he developed, he showed that GAs are able to find the best known solution for this structure, if the atoms are allowed a small relaxation ($\pm 0.4\mathring{A}$) from the best known solution. In addition, the categorical variables were restricted to the best known value; otherwise GAs would generate a large number of invalid structures, i.e., physically unacceptable solutions [13].

Our present work continues the effort to find a global optimizer for the LEED surface structure determination problem. We have adapted and applied pattern search methods to solve this optimization problem and we were able to verify that pattern search methods have better performance than GAs. They generate significantly fewer invalid structures, and they are able to find the best known solution several times faster than GAs under the same local relaxations. The new methods can also find the best known solution and even improved solutions under complete relaxation of the atoms. Finally, the new methods allow for the simultaneous search over the whole parameter space (continuous variables and categorical variables), which is essential for a method to be used as a global optimizer in the LEED surface structure determination problem. Our numerical experiments also show that pattern search methods are robust.

Pattern search methods have been widely used in many applications. One version, the Hooke and Jeeves (HJ) search algorithm [18] was used to solve the LEED surface structure problem for CdTe(110) [19, 20]. However, this approach required an initial search that used heuristics to find good initial points near some promising local minima before running HJ algorithm, where it was used as a local optimizer for the continuous variables. Since then, there have been many improvements, both theoretically and algorithmically, to the class of pattern search methods. In particular, methods for handling constraints and categorical variables have been developed. These new developments allow us to apply them to the global LEED surface structure determination problem. We will review these methods in more detail in Section 2.

The remaining sections of the paper are organized as follows. In Section 2 we review the pattern search method, with particular attention to its convergence properties. Section 3 describes how to apply pattern search methods to the LEED surface structure problem. In Section 4 we present our numerical results on one test problem, the complex surface structure determination of Ni-(5x5)-Li, and compare our results with those of genetic algorithms. In Section 5 we give concluding remarks.

2 Pattern Search Methods

Pattern search methods belong to a class of optimization methods known as direct search methods. These methods have a long and rich history in the scientific and engineering communities where they have been applied to numerous problems. An excellent introduction and survey of these methods can be found in [21], which also contains numerous references. The main attraction of direct search methods is their ability to find optimal solutions without the need for computing derivatives in contrast to the more familiar gradient-based methods. This feature is particularly attractive in situations where the objective function is a result of a simulation code.

The basic ideas behind these methods can be demonstrated through one of the early pattern search methods known as compass search. The algorithm can be summarized as follows: from the current point, we try steps in each of the four compass directions, East, West, North, and South. If one of these four steps yields a reduction in the function, the improved point becomes the new iterate; if none of these steps yields an improvement, we try again with steps that are half as large.

Fig. 2 shows the first 5 iterations of the pattern search method applied to a typical minimization problem. The algorithm starts with a finite initial step length. As the iterate approaches the solution, the algorithm reduces the length of the steps (this turns out to be central to the convergence analysis). Eventually, when the step length falls below a certain tolerance, the algorithm is said to have converged and the search stops.

As this example demonstrates, an appealing characteristic of the pattern search method is that it is simple and easy to implement and it only requires the ability to evaluate the function at a point. A known disadvantage is that although the method may quickly decrease the initial function value, it may be slow to detect a minimizer due to the convergence test being dependent on reducing the step length below a certain threshold. Unfortunately, this is the price of not explicitly using any derivative information.

2.1 The algorithm and convergence analysis

Several generalizations of the simple pattern search method have been proposed and discussed previously, see for example [21, 23]. These methods fall under the general classification of generating set search (GSS) methods, which are categorized by using multiple search directions computed from a generating set. One example from this class of methods is the generalized pattern search (GPS) method [24]. This method, originally developed for unconstrained optimization problems, has been extended by several authors to include general nonlinear constraints and the use of mixed variables. This latter property in particular is essential to our use of the algorithm for the LEED surface structure determination problem. In addition, the ability to specify general constraints is useful, when one wishes to impose physical restrictions on possible solutions, for example symmetry or realistic bond distances. For details about the GPS method

for constrained optimization, see sections 7 and 8 of [21]. The GPS method for mixed variable problems is described in more detail in [25].

Consider the following problem:

$$\text{minimize } f(x)$$

where $x \in \mathcal{R}^n$, $f : \mathcal{R}^n \to \mathcal{R}$ ($\mathcal{R}^n$ denotes the n-dimensional real search space).

We define a generating set (positive spanning set), $\mathcal{D}$, as a set of vectors whose nonnegative linear combinations span $\mathcal{R}^n$. For instance, a positive spanning set $\mathcal{D}$ for $\mathcal{R}^n$ could be

$$\{e_1, e_2, \cdots, e_n, -e_1, -e_2, \cdots, -e_n\},$$

where e_i is the i-th unit Cartesian vector in $\mathcal{R}^n$. We note that this set must contain at least $n+1$ vectors to guarantee nonnegative linear combinations and hence need not be unique. The GPS method will take steps through search space comparing function values at each of the points defined by one of the search directions and step lengths. We will let Δ_k be the step length control parameter, and let Δ_{tol} be the tolerance used to test for convergence. Assume that the algorithm starts with an initial guess x_0 that has a finite function value and an initial step length $\Delta_0 > \Delta_{\text{tol}}$. Then the GPS method can be described as follows:

Algorithm 1: Generalized Pattern Search

1: Choose generating set $\mathcal{D}$, for example, let $\mathcal{D} = \{e_1, e_2, \cdots, e_n, -e_1, -e_2, \cdots, -e_n\}$
2: Choose Δ_0
3: **for** $k = 1, 2, \ldots$ **do**
4: **if** there exists $d_k \in \mathcal{D}$ such that $f(x_k + \Delta_k d_k) < f(x_k)$ **then**
5: Set $x_{k+1} = x_k + \Delta_k d_k$ ▷ update the iterate.
6: Set $\Delta_{k+1} = \Delta_k$ ▷ no change to the step length control parameter
7: **else**
8: Set $x_{k+1} = x_k$ ▷ $f(x_k + \Delta_k d_k) \geq f(x_k)$ for all $d_k \in \mathcal{D}$; do not update the iterate
9: Set $\Delta_{k+1} = \frac{1}{2}\Delta_k$ ▷ contract the step length control parameter
10: **if** $\Delta_{k+1} < \Delta_{\text{tol}}$ **then**
11: GPS algorithm has converged
12: **end if**
13: **end if**
14: **end for**

Each step of the GPS algorithm can be generalized even further. For example, in Step 1, the lengths of the vectors in the generating set can take on any values between specified lower and upper bounds; also, a finite number of additional search directions (other than the ones already included in the generating set), may be added using heuristics or any other physics-based approach that seems appropriate; for example, random search, a few generations of a genetic algorithm, or Latin hypercube search. This adds an optional search step in each iteration of a GPS algorithm. The search along the directions of the generating set is commonly referred to as a local poll step. In Step 4, instead of requiring simple decrease of the function value, one may instead require a larger decrease. Finally, different scale factors may be used to update the step length control parameter Δ_k, so that it is not always 1 in Step 6 and 1/2 in Step 9. These

generalizations allow for great latitude in using the GPS method and can have a significant impact on the efficiency of the algorithm.

One of the main advantages of the GPS method is that, although derivative information is not explicitly used, the method can be shown to have strong convergence properties, which are in many cases as good as for methods that require derivative information. This fact was not fully appreciated until Torczon showed convergence of these algorithms under mild conditions and Audet later proved that the convergence results could not be strengthened without adding additional constraints [24]. The general outline of the proof can be shown to rely on satisfying two conditions: the GPS algorithm needs to pick a descent direction, and the GPS algorithm must avoid poor step lengths. The idea of the generating set enforces the first condition, and the second condition is enforced through shortening the step length at unsuccessful iterates. The surprising aspect of this class of algorithms is that convergence can be shown even though the derivative of the function is never explicitly computed.

For our results, we have used a particular implementation of the GPS algorithm which is part of a software package, NOMADm [26]. NOMADm is a MATLAB implementation of the class of Mesh-Adaptive Direct Search (MADS) filter algorithms, which is a generalization of pattern search, for numerically solving nonlinear and mixed variable optimization problems with general nonlinear inequality constraints. Since it is not a straightforward task to generalize the concepts of a direct search algorithm to categorical variables, many available implementations of GPS methods (including both parallel and serial codes) are specifically designed for problems with continuous variables only. By contrast, NOMADm allows categorical variables, which has a distinct advantage for our problem. Due to the difficulty posed by categorical variables, there are no search or polling strategies suggested for categorical variables in NOMADm either; nonetheless it provides an interface from which user-defined strategies can be applied to the categorical variables. In our results, for categorical variables, at each iteration we have used an optional search step adapted from Latin hypercube search for continuous variables [26] and a "generating set" adapted from the set $\{e_1, e_2, \cdots, e_n, -e_1, -e_2, \cdots, -e_n\}$ for continuous variables for a local poll step.

3 LEED Surface Structure Determination Using GPS

Low-energy electron diffraction (LEED), in its most common application, is used in an iterative procedure to predict surface structures. The LEED experiment is first simulated with a sophisticated scattering model. For accuracy, the simulation must include multiple scattering, which makes the computation expensive especially for complex structures. Since the computation can only be performed with a set of given atomic positions and chemical identities, it is necessary to start by guessing the structure and then iterating the search through the space of physically realistic solutions.

The distinction between local and global optimization is important here. Efficient methods of local optimization are available for LEED, as long as there are no categorical variables: for example, tensor LEED [27], which provides local derivatives with respect to the continuous variables (atomic coordinates). Tensor LEED can also assist with categorical variables (the atomic identities) in a method sometimes called "chemical Tensor LEED" [28, 29] for surface structures structurally close to a certain reference. By contrast, global optimization is much more demanding. The larger challenge lies in solving the optimization problem in the presence

of many other local minima. The LEED data itself does not provide much information to determine where these other minima may be, so an exhaustive search (for both continuous and categorical variables) would appear to be necessary resulting in a computationally expensive solution. The present work focuses on the need to make such global optimization in LEED as efficient as possible.

The main output from a LEED code is the mismatch between the simulation and the experimental data, which is encapsulated in a number called the R-factor [30]. In principle, the R-factor is a continuous function of atomic positions, but it is discontinuous in terms of atomic identities (one may also conveniently view a vacancy as just another atomic type, with vanishing scattering strength). In practice, the functional dependence of the R-factor on the continuous variables is not perfectly smooth, due to numerical discreteness in the experiment and computation (e.g. typical computations involve iterations which are interrupted at variable numbers of iterations, giving discontinuous results, similarly, theory includes an adjustable energy shift which often causes discontinuous changes of the end points of the energy range common with experiment).

Fortunately, there are powerful methods that reduce the solution space to be searched and therefore the required computing effort. One is symmetry: many surface structures turn out to be symmetrical relative to rotation, mirror or glide operations. Symmetry helps in two major ways: the computation can be considerably accelerated (sometimes by orders of magnitude) and memory usage is reduced. At the same time, the dimensionality of the search space is greatly reduced. For these reasons, such symmetry is extensively used in our LEED codes [31].

The other computational savings come from imposing "realism" on the search space. At a simple level, one can impose upper and lower bounds on the interatomic distances (bond lengths) and on the coordination numbers (numbers of nearest neighbors). This is commonly done in organic crystallography with x-ray diffraction, but rare for the remainder of the periodic table, because the rules for realism depend strongly on the nature of interatomic bonding, which varies greatly among pairs of elements. In particular, this constraining of the search has not yet been applied with LEED, except in preliminary explorations with genetic algorithms [13]. On the other hand, it has been almost universal in LEED studies to sharply constrain the chemical elements: usually the experimenter knows which elements are present (although not necessarily where), so this can be justified.

Our intent was thus to investigate the applicability and performance of a generalized pattern search method for determining the surface structure of materials with LEED. In our calculations, we used the Symmetrized Automated Tensor LEED package [31] to calculate the R-factor, and used the package NOMADm [26] for the pattern search methods as mentioned in Section 2. As in Ref. [13], we applied the same set of necessary constraints to the search, and the structures that don't satisfy these constraints (invalid structures) were assigned a big R-factor (e.g., 1.6). The atomic positions can be expected to be obtained with an accuracy better than $0.1\mathring{A} = 0.01$nm, based on experience with extended surfaces [4].

4 Numerical Results and Discussion

We applied the GPS method to the complex surface formed by adding lithium (Li) atoms on a nickel (Ni) crystal surface with (001) crystallographic orientation, and a (5x5) superlattice; it is commonly labeled as Ni(001)-(5x5)-Li. This structure was analyzed by LEED with only the

benefit of local optimization as provided by tensor LEED [27]. To perform the global search, the authors of that analysis conceived of 45 different structural models from physical information for this surface (such as symmetry, bond lengths and density of the Li layer). Each model was locally optimized by tensor LEED; the best-fit model has the lowest R-factor value 0.24 (see Fig. 3). We refer to this structure henceforth as the best known structure for this surface. This structure can be viewed as having a pure nickel substrate, covered with three overlayers, which have, in each (5x5) unit cell, 25 Ni atoms in the innermost overlayer, 16 Ni and 8 Li in the middle overlayer, and 9 Li atoms in the outermost overlayer. The superlattice cell has a size of about $12.5\mathring{A} \times 12.5\mathring{A}$, and the three layers have a combined thickness (in the assumed models) of about $5\mathring{A}$. If the full point group symmetry of the Ni (001) substrate (namely p4m) is imposed, there are 14 inequivalent atoms in the cell (for example, those atoms located within one octant). Therefore, to determine the structure of this surface, we need only the coordinates of these 14 atoms as well as their chemical identities. Correspondingly, the optimization problem has $3 \times 14 = 42$ continuous variables with a parameter space volume of $12.5\mathring{A} \times 12.5\mathring{A} \times 5\mathring{A}$ and 14 categorical variables (each atom can be either Ni or Li). There is one non-structural parameter–an adjustable "muffin-tin zero" energy shift due to an unknown energy zero in the theory relative to experiment, and the R-factor is an optimal value also with respect to this energy shift. We did not count it as an optimization variable here, since this optimization is done inside the TLEED code, uncontrolled by the GPS code. To easily permit other models in this scheme one could allow the categorical variables to also represent a vacancy, so that the number of atoms can vary, but this was not done in this investigation.

To compare the performance of the GPS method to that of GAs, we performed similar calculations on this test problem using both approaches. As mentioned in the introduction, due to the large number of invalid (i.e. physically unrealistic) structures produced, only small atomic relaxations ($\pm 0.4\mathring{A}$) from the best known solution were allowed in the search with GAs. Fig. 4 illustrates the performance of the GAs: the horizontal axis shows the number of function evaluations, while the vertical axis shows the R-factor. In this example, the atomic types (chemical identities) are fixed to those of the best known structure. Since the initial guesses for the atomic positions were restricted within $\pm 0.4\mathring{A}$ from the best known atomic positions, this example actually represents near-local rather than global optimization. The best known structure was found at around 2400 function evaluations.

The same calculation as in Fig. 4 was performed by the GPS method,with results shown in Fig. 5. From these two figures, we can see that GPS requires fewer function calls to locate an optimal solution. GPS found an improved solution (R-factor=0.12) at around 600 function evaluations, while the GA found the best known solution (R-factor=0.24) already at around 2400 function evaluations. Another observation from the figures is that GPS generates far fewer invalid structure than the GA. The first 1000 trial structures produced by GA were all invalid, and after the first valid trial structure was found, a large fraction of the trial structures remained invalid. By contrast, the first 100 trial structures generated by GPS were all invalid, but once a valid structure was found, GPS generated an increasingly large fraction of valid structures until no more invalid structures were produced.

In our next test, we kept the atomic positions (42 continuous variables) fixed, only allowing the chemical identities (categorical variables) to change. Fig. 6 and Fig. 7 show results by GA and GPS, respectively. Both methods started from the same initial guess of the categorical variables, labeled as 11111122211122; in this notation each of the 14 adjustable atoms is listed as having chemical type 1 for Li or 2 for Ni, with the atoms listed in some predefined but arbitrary

order. From these two figures, we see that GA needs 280 function evaluations to obtain the best known solution 11111222222222, while GPS needs only 49 function evaluations; moreover, GPS found an improved solution at 135 function evaluations. Again, we see that in these tests GPS requires far fewer function evaluations than GA. It should be noted for completeness, however, that the improved structure found by GPS with the R-factor near 0.12 is actually also invalid, as it is found to have an unphysically large shift of the energy zero; this nonetheless does not reflect negatively on the GPS method, but points to the need for better limits on the allowable range of this variable.

To test the robustness of the GPS method, we ran the same search as above with 20 different randomly chosen initial guesses: Fig. 8 shows the results of all 20 trials. We see that the GPS method is very robust: all 20 runs reached the same minimum R-factor of 0.1184 with an average of 152 function evaluations.

To be used as a global optimizer in the LEED surface structure determination problem, one of the key requirements of the method is the ability to *simultaneously* relax both the continuous and categorical variables; another requirement, of course, is the ability to relax atomic positions globally (not just locally), so the search for the optimum really explores the whole parameter space. Fig. 9 shows that GPS allows relaxing both kinds of variables simultaneously. In this calculation, we started with the best known atomic positions with a random displacements with a window size $\pm 0.4\mathring{A}$ and a random guess of the chemical identities, and we completely relaxed atoms in the xy plane, while we kept only the layer constraints in the z direction (this limitation was chosen mainly for the sake of simplicity of programming; there is no difficulty in principle to remove this constraint). We can see from Fig. 9 that after 650 function calls both the best known categorical and continuous variables were found. Further searching found an improved solution with R-factor=0.2275 at around 660 function calls.

5 Conclusions

The generalized pattern search (GPS) method has been applied to the surface structure determination of the complex structure Ni(001)-(5x5)-Li by low energy electron diffraction (LEED). Our evaluation shows that the GPS method exhibits a better performance than the previously used genetic algorithm (GA) in terms of efficiency and ability to locate the optimal solutions, at least for the relatively local search that the earlier GA tests were limited to. In addition, GPS produces far fewer invalid (i.e. unphysical) structures and requires far fewer function evaluations than GA.

We were also able to demonstrate that the GPS method is robust and allows complete relaxation of the atomic positions, as required for global optimization. Moreover, it allows simultaneously relaxing both the atomic positions and atomic types (chemical identities). We also found solutions with a better fit over the previously best known solution (even though the energy zero shift of one of the solutions is unrealistic for reasons unrelated to GPS). Our evaluation shows that GPS optimization methods are promising candidates for LEED surface structure determination.

It is clear that for GPS, as for other optimization methods, a validity test would be very useful to further reduce the time-consuming calculation of unphysical structures generated during the optimization process. Such a test could include limits on interatomic distances and angles, as well as on the unknown energy zero shift between theory and experiment. It could even, in a

more sophisticated approach, include the total energy of the system, since that energy must also be a minimum for the correct solution. These are all subjects for future research.

Acknowledgments

This work would not have been possible without the support of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231. The authors would like to thank Drs. Chao Yang, Lin-Wang Wang, Xavior Cartoixa Soler, Byounghak Lee, Andrew Canning and other members of nano-science project in the Lawrence Berkeley National Lab for many useful discussions and specific help with the software package. The numerical experiments used resources of the National Energy Research Scientific Computing Center.

References

[1] D. P. Woodruff and T. A. Delchar, "Modern Techniques of Surface Science," 1986, Cambridge University Press

[2] M. Henzler and W. Göpel, "Oberflächenphysik des Festkörpers," 1991, Stuttgart, Teubner Verlag

[3] G. Ertl and J. Küppers, "Low Energy Electrons and Surface Chemistry", 1985, Weinheim, Verlag Chemie

[4] M. A. Van Hove, W. H. Weinberg and C.-M. Chan, "Low Energy Electron Diffraction," 1986, Berlin, Springer Series in Surface Sciences 6

[5] K. Heinz, Prog. Surf. Sci. 27 (1988) 239

[6] P. J. Rous, Surf. Sci., 296 (1993) 358.

[7] V.B. Nascimento, V.E. de Carvalho, C.M.C. de Castilho, E.A. Soares, C. Bittencourt, D.P. Woodruff, Surf. Rev. Lett. 6, 651 (1999).

[8] E.A. Soares, V.B. Nascimento, V.E. de Carvalho, C.M.C. de Castilho, A.V. de Carvalho, R. Toomes, D.P. Woodruff, Surf. Sci. 419, 89 (1999).

[9] V.B. Nascimento, V.E. de Carvalho, C.M.C. de Castilho, B.V. Costa, E.A. Soares, Surf. Sci. 487, 15 (2001).

[10] E.D. Correia, V. Nascimento, C.M.C. de Castilho, A.S.C. Esperidiao, E.A. Soares, V.E. de Carvalho, J. Phys. Cond. Matt. **17**, 1 (2005).

[11] M. Kottke, K. Heinz, Surf. Sci. 376, 352 (1997).

[12] R. Döll and M. A. Van Hove. Surf. Sci. **355**, L393(1996).

[13] G. S. Stone, M.S. Dissertation, Computer Science Department, San Francisco State University, 1998.

[14] J. H. Holland, Adaptation in Natural and Artificial Systems (The University of Michigan Press, Ann Arbor, MI, 1975).

[15] D. E. Goldberg. Genetic Algoriths in Search, Optimization and Machine Learning (Addison-Wesley, Reading, MA, 1989).

[16] L. Davis, Ed., Handbook of Genetic Algorithms (Van Nostrand Reinhold, New York, 1991).

[17] H. Jiang, S. Mizuno and H. Tochihara, Surf. Sci. 380, L506 (1997)

[18] R. Hooke and T.A. Jeeves, J. Assoc. Computing Machinery 8 (1961) 212.

[19] P.G. Cowell and V.E. De Carvalho, Surf. Sci. 187 (1987) 175

[20] P.G. Cowell and V.E. De Carvalho, J. Phys. C: Solid State Phys. **21** (1988) 2983

[21] T. G. Kolda, R. M. Lewis, V. Torczon, SIAM Review, vol 45, No. 3 pp. 385 (2003).

[22] C. G. Broyden, A class of methods for solving nonlinear simultaneous equations, Math. Comp., 19 (1965), pp. 577.

[23] V. Torczon, SIAM J. Opt., vol 7, pp. 1 (1997).

[24] Charles Audet, Technical Report CRPC-TR98779, Rice University, (1998)

[25] Charles Audet and J.E. Dennis, Jr., SIAM J. Optim., **11** (3), 573-594 (1999)

[26] M. Abramson, NOMADm software http://en.afit.edu/ENC/Faculty/MAbramson/NOMADm.html

[27] M. A. Van Hove, W. Moritz, H. Over, P. J. Rous, A Wander, A. Barbieri, N. Materer, U. Starke, G. A. Somorjai, Surf. Sci. Rep, 19 (1993) 191

[28] K. Heinz, Rep. Progr. Phys. 58, 637 (1995);

[29] K. Heinz, R. Döll and M. Kottcke, Surf. Rev. and Lett. 3, 1651 (1996).

[30] J.B. Pendry, J. Phys **C13** (1980) 937.

[31] A. Barbieri and M.A. Van Hove, *Symmetrized Automated Tensor LEED package*, http://www.sitp.lbl.gov/index.php?content=/leedpack/leedpack.html.

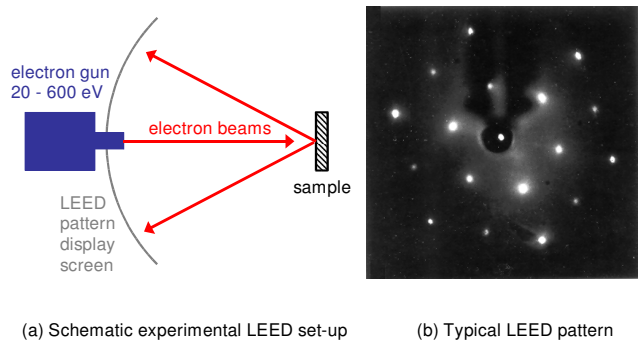

Figure 1: A typical LEED experiment set-up (a) and a typical LEED pattern (b).

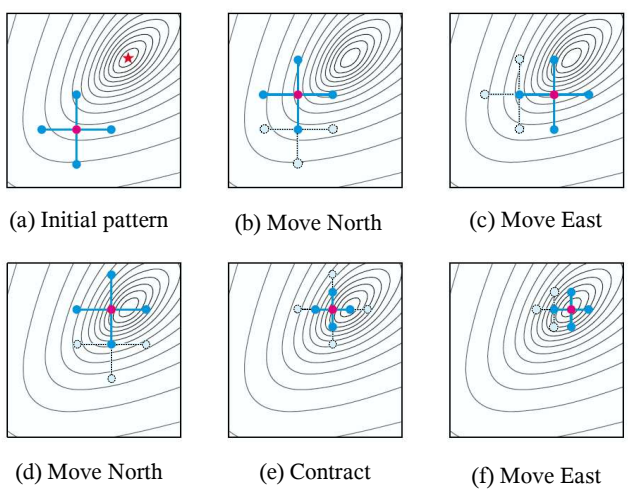

Figure 2: First 5 iterations of the pattern search method applied to a 2-dimensional optimization problem (Broyden tridiagonal function [22]). The gray lines in the background are level sets, and the solution to the problem is marked with a star. In each panel, the center dot denotes the current iterate, and the 4 surrounding dots represent the trial points under consideration at that iteration; the trial points from the previous iteration are shown in dotted circles for comparison. The algorithm starts with a certain step length; when the iterate approaches the solution, the algorithm reduces the length of the steps taken; when the step length falls below a certain tolerance, the algorithm stops. This example is from Ref. [21].

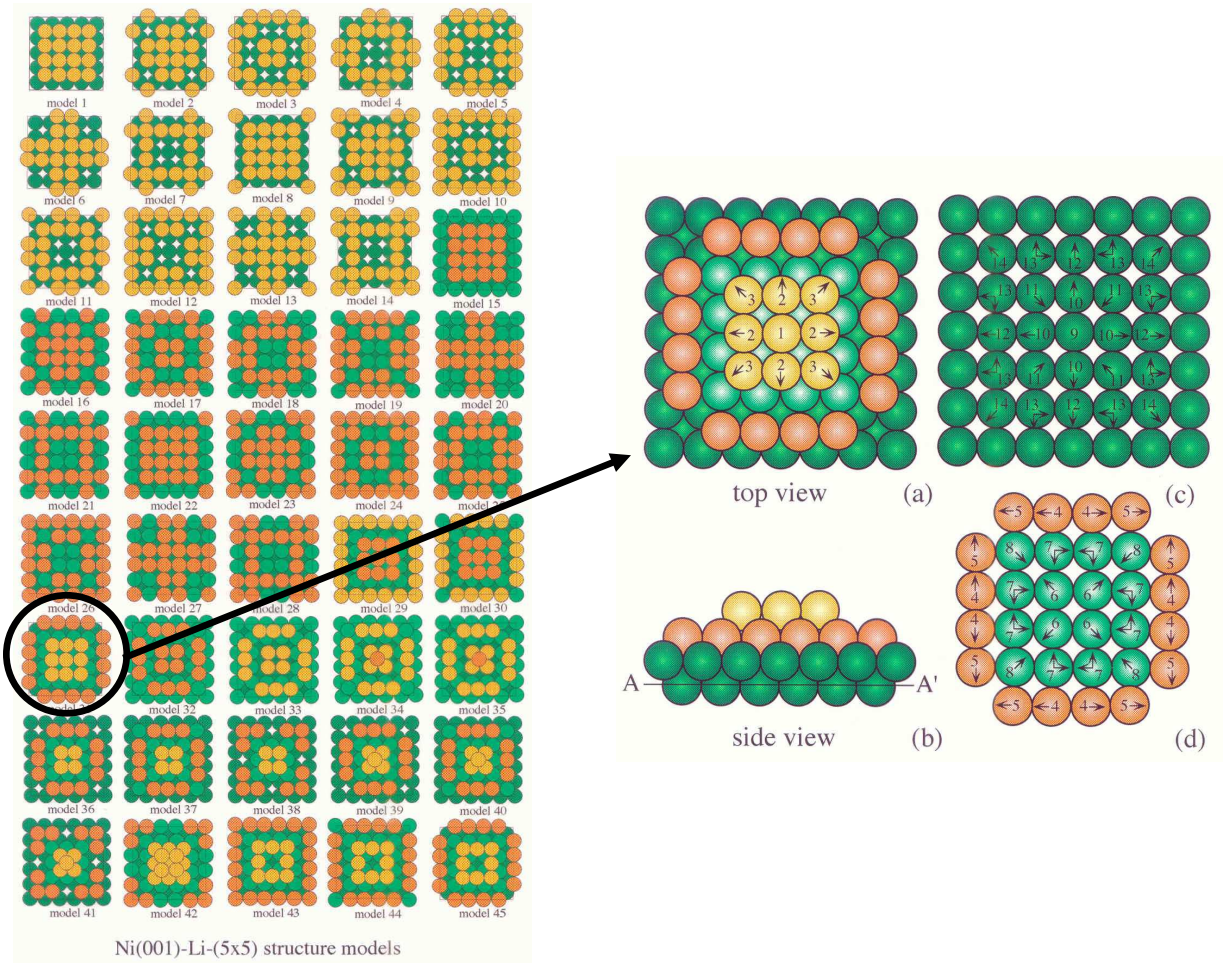

Figure 3: The surface formed by adding lithium atoms on a nickel surface. The 45 small panels on the left are structure models built to fit LEED experimental intensities from physical intuition (each model was locally optimized by tensor LEED). The 4 panels on the right detail structure model 31 (panels (c) and (d) represent two separate layers), which has the lowest R-factor among these models: we label it the "best known solution" (Courtesy of Dr. S. Mizuno) [17].

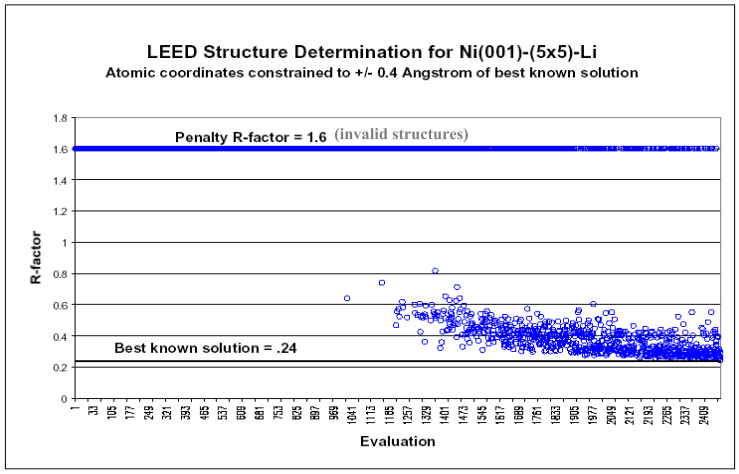

Figure 4: A GA result of continuous variable search following Ref. [13]. Only the continuous variables are relaxed; the categorical variables are fixed at the best known chemical identities. The best known structure was found after about 2400 function calls. Note: the relaxation of the atoms was constrained to be $\pm 0.4\mathring{A}$ from the best known atomic positions.

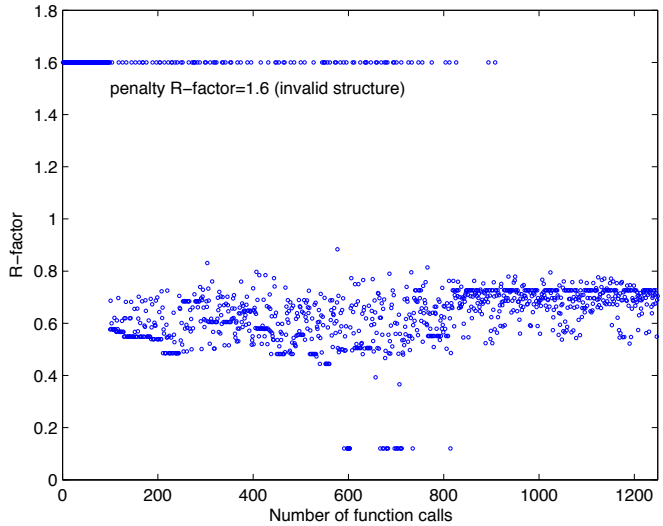

Figure 5: As Fig. 4, but using GPS, which generates fewer invalid structures than GA. At 591 function evaluations the improved solution was found (R-factor=0.12).

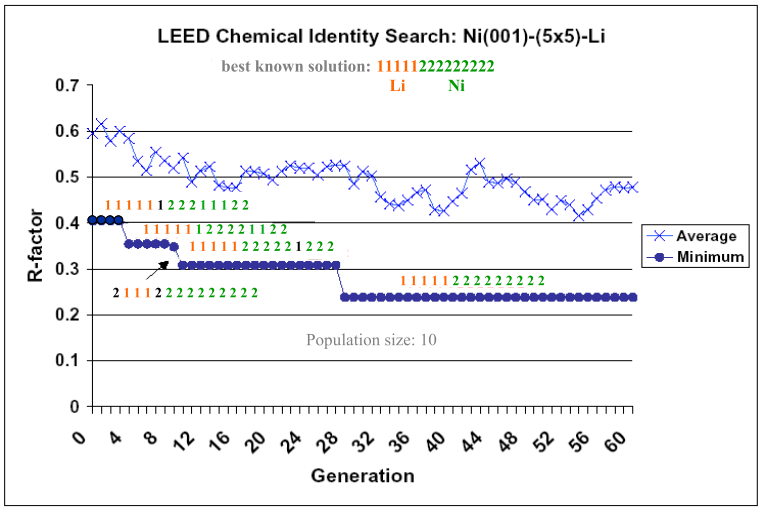

Figure 6: A GA result of chemical identity search following Ref. [13]. Only the categorical variables are relaxed; continuous variables are fixed at the best known atomic positions. The best known solution was found at 280 function evaluations. Note that the number of function evaluations in GAs is the product of the number of generations and the population size (10).

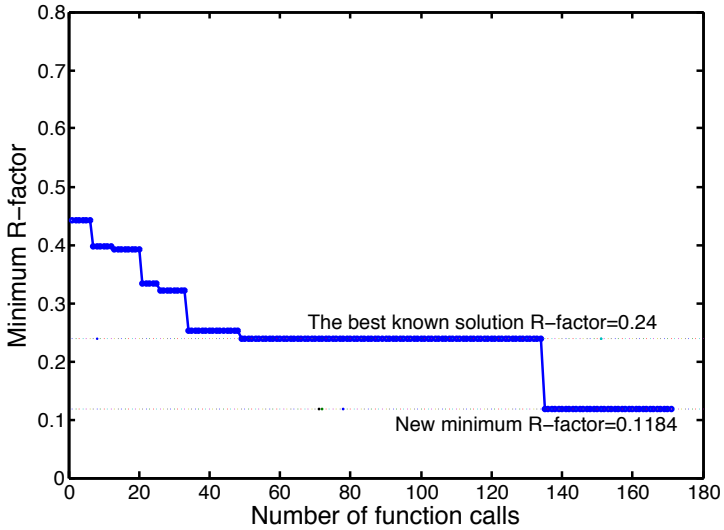

Figure 7: As Fig. 6, but using GPS, which requires fewer function evaluations than GA. At the 49th function call, the best known solution was found; at the 135th function call, an improved solution with R-factor=0.1184 was found.

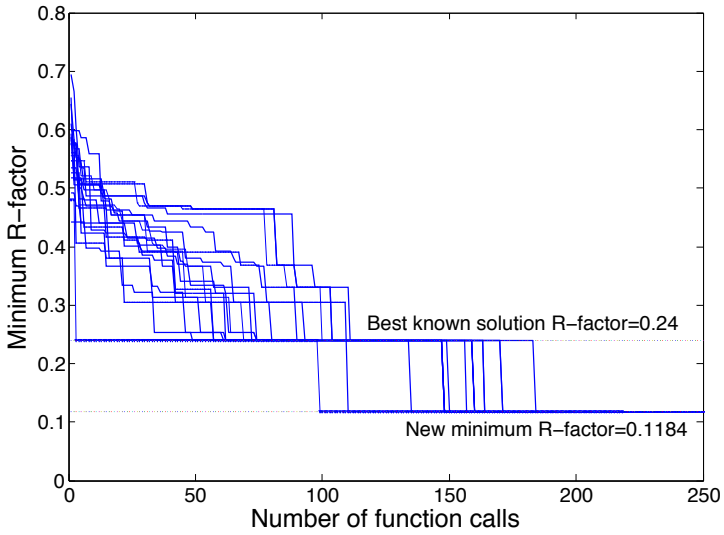

Figure 8: Same GPS search as in Fig. 7, with 20 different randomly chosen initial guesses. This shows that GPS is robust. All 20 runs of the chemical identity search reached the same minimum R-factor.

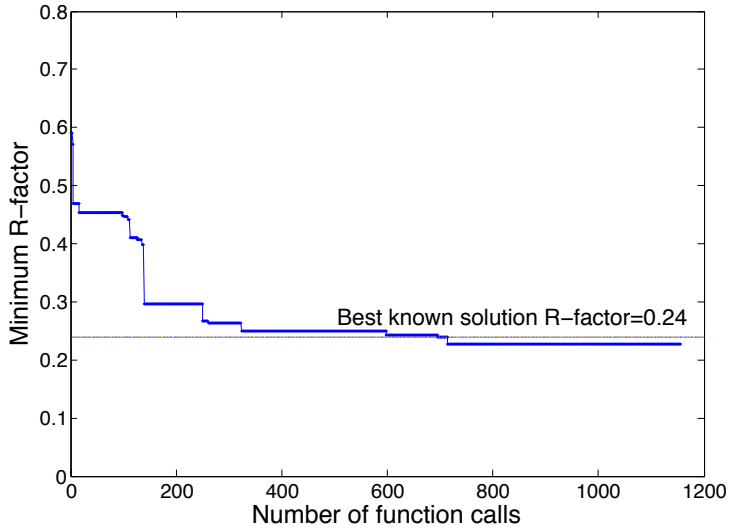

Figure 9: A GPS run starting with the best known atomic positions with random displacements, with a window size of $\pm 0.4\mathring{A}$ and a random guess of the chemical identities, allowing relaxations of both types of variables: complete relaxation of atomic positions in the in-plane x and y directions, and layer constraints in the z direction. An improved solution with an R-factor=0.2275 was found.