Optimization methods for finding minimum energy paths

Daniel Sheppard, Rye Terrell, and Graeme Henkelman[a)]
*Department of Chemistry and Biochemistry, The University of Texas at Austin,
Austin, Texas 78712-0165, USA*

A comparison of chain-of-states based methods for finding minimum energy pathways (MEPs) is presented. In each method, a set of images along an initial pathway between two local minima is relaxed to find a MEP. We compare the nudged elastic band (NEB), doubly nudged elastic band, string, and simplified string methods, each with a set of commonly used optimizers. Our results show that the NEB and string methods are essentially equivalent and the most efficient methods for finding MEPs when coupled with a suitable optimizer. The most efficient optimizer was found to be a form of the limited-memory Broyden-Fletcher-Goldfarb-Shanno method in which the approximate inverse Hessian is constructed globally for all images along the path. The use of a climbing-image allows for finding the saddle point while representing the MEP with as few images as possible. If a highly accurate MEP is desired, it is found to be more efficient to descend from the saddle to the minima than to use a chain-of-states method with many images. Our results are based on a pairwise Morse potential to model rearrangements of a heptamer island on Pt(111), and plane-wave based density functional theory to model a rollover diffusion mechanism of a Pd tetramer on MgO(100) and dissociative adsorption and diffusion of oxygen on Au(111). © *2008 American Institute of Physics*. [DOI: 10.1063/1.2841941]

I. INTRODUCTION

Computational methods for calculating minimum energy paths (MEPs) are widely used in the fields of theoretical chemistry, physics, and materials science. The MEP describes the mechanism of reaction, and in thermal systems, the energy barrier along the path can be used to calculate the reaction rate.

Here, we compare several approaches for finding MEPs. We have restricted our investigation to methods in which the initial and final states are known. Our goal is then to find the MEP between these states to a specified accuracy with the smallest amount of computational effort.

This paper is structured in the following way. In Sec. II, the nudged elastic band (NEB) method[1–3] is summarized. In Sec. III, we state our objectives and convergence criteria for finding MEPs. In Sec. IV, we describe the optimizers that we use to converge the NEB. Convergence results are presented in Sec. V for a model system of island rearrangement on a (111) surface using a pairwise Morse potential.

Using the NEB as a baseline, recently developed methods are compared in subsequent sections. These include the doubly nudged elastic band (DNEB) (Ref. 4) (Sec. VI), the string[5] (Sec. VII), and the simplified string[6] (Sec. VIII) methods.

Finally, we reproduce our NEB optimizer tests for three different surface reactions modeled with density functional theory: in Sec. IX A, Pd_4 diffusion on MgO(100), and in Sec. IX B, O_2 dissociative adsorption and O diffusion on Au(111).

II. NUDGED ELASTIC BAND METHOD

The NEB is a method to find a MEP between a pair of stable states.[1] In the context of reaction rates, this pair has an initial and a final state, both of which are local minima on the potential energy surface. The MEP has the property that any point on the path is at an energy minimum in all directions perpendicular to the path. This path passes through at least one first-order saddle point. The MEP can also be described as the union of steepest descent paths from the saddle point(s) to the minima.

The NEB is a chain-of-states method[7,8] in which a string of images (geometric configurations of the system) is used to describe a reaction pathway. These configurations are connected by spring forces to ensure equal spacing along the reaction path. Upon convergence of the NEB to the MEP, the images describe the reaction mechanism, up to the resolution of the images (see Fig. 1).

A NEB calculation is started from an initial pathway connecting initial and final states. Typically, a linear initial path is sufficient, but in some cases, a different choice is better. For example, if atoms get close to each other along the linear path, a geometric repulsive force can be used to push these atoms apart, resulting in a band with lower initial forces. An interpolation in internal coordinates can also yield a more suitable initial pathway,[9] for example, if the reaction involves rotational motion. When the reaction is known to go through an intermediate state, an initial path can be constructed from segments through the intermediate.

The images along the NEB are relaxed to the MEP through a force projection scheme in which potential forces act perpendicular to the band, and spring forces act along the band. To make these projections, the tangent along the path $\hat{\tau}$

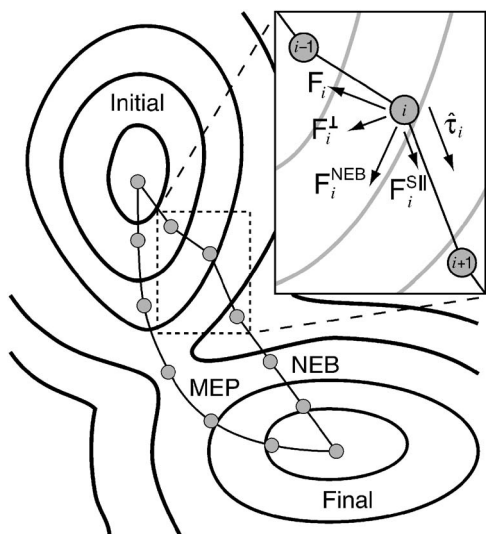a)Electronic mail: henkelman@mail.utexas.edu.

FIG. 1. Two components make up the nudged elastic band force $\mathbf{F}^{\text{NEB}}$: the spring force $\mathbf{F}_i^{S\|}$, along the tangent $\hat{\boldsymbol{\tau}}_i$, and the perpendicular force due to the potential $\mathbf{F}_i^{\perp}$. The unprojected force do to the potential $\mathbf{F}_i$ is also shown for completeness.

is defined as the unit vector to the higher energy neighboring image.[2] A linear interpolation between the vectors to neighboring images is used at extrema so that the direction of $\hat{\boldsymbol{\tau}}$ does not change abruptly. This upwinding tangent improves the stability of the NEB and avoids the development of artificial kinks in high force regions along the path.[2] The NEB force on image i contains two independent components,

$$\mathbf{F}_i^{\text{NEB}} = \mathbf{F}_i^{\perp} + \mathbf{F}_i^{S\|}, \tag{1}$$

where $\mathbf{F}_i^{\perp}$ is the component of the force due to the potential perpendicular to the band,

$$\mathbf{F}_i^{\perp} = -\nabla(\mathbf{R}_i) + \nabla(\mathbf{R}_i) \cdot \hat{\boldsymbol{\tau}}_i \hat{\boldsymbol{\tau}}_i, \tag{2}$$

and $\mathbf{F}_i^{S\|}$ is the spring force parallel to the band,

$$\mathbf{F}_i^{S\|} = k(|\mathbf{R}_{i+1} - \mathbf{R}_i| - |\mathbf{R}_i - \mathbf{R}_{i-1}|)\hat{\boldsymbol{\tau}}_i. \tag{3}$$

In this final expression, $\mathbf{R}_i$ is the position of the ith image and k is the spring constant.

The saddle point is particularly important for characterizing the transition state within harmonic transition state theory (TST). The difference between the saddle point energy and that of the initial state determines the exponential term in the Arrhenius rate, and the MEP can be obtained by minimizing from the saddle point(s). An efficient strategy for finding a saddle between known states is to roughly optimize a NEB calculation and then do a 'min-mode' following saddle point search[10,11] from the highest energy image to find the transition state.[2,4] Another approach, which avoids having to run two separate optimizations or interpolate to find the saddle, is the climbing-image NEB (CI-NEB).[3] In this method, the highest energy image l feels no spring forces and climbs to the saddle via a reflection in the force along the tangent,

$$\mathbf{F}_l^{\text{CI}} = \mathbf{F}_l - 2\mathbf{F}_l \cdot \hat{\boldsymbol{\tau}}_l \hat{\boldsymbol{\tau}}_l. \tag{4}$$

Once the saddle is found, the normal mode frequencies can be calculated to ensure that the saddle is first order and to find the prefactor of the reaction.

III. CONVERGENCE OBJECTIVES

The methods tested here are evaluated using the following optimization objectives and convergence criteria. Our principal goal is to compare methods as they are used by researchers who do calculations of reaction pathways in chemistry and materials science. Most of these methods have parameters that can be tuned to optimize performance. Our approach is to determine a set of near-optimal parameters which can be used for all tests, rather than reoptimize the parameters for each calculation. This strategy reflects the way computational methods are normally used; there is little advantage to a method that requires significant additional calculations for parameter tuning.

Our objective is to find minimum energy pathways between stationary states. Specifically, how this objective is defined, however, can lead to different conclusions about which methods are efficient. Thus, we need to clarify exactly what we are interested in calculating and emphasize that our tests reflect the bias of chemistry and material science, in which MEPs are calculated to find both the mechanism and activation energy of reactions.

Generally, a researcher wants to understand the mechanism (the geometric pathway) of reaction qualitatively and the activation energy with higher accuracy. By a qualitative reaction pathway, we mean that intermediate minima along the pathway should be identified by the MEP search. By an accurate activation energy, we mean that, in principle, the saddle point energy could be found to arbitrary accuracy on a given potential energy surface, but in practice, it should be known well enough to give an accurate harmonic TST rate. In these tests, a climbing-image approach is used so that one image along the band is converged to the saddle point to a specified precision.[3]

In this study, we investigate single step reaction mechanisms where eight or fewer images can be used to resolve the MEP. The number of images was kept fixed in our tests, even though fewer images could be used to resolve the MEP and find the saddle point at lower computational cost. Using the same philosophy as applied to optimizer parameters, we want to use a conservative number of images that would identify intermediate minima between nearby initial and final states. The limited resolution along the path will result in some deviation between the images and the true MEP. This is not a disadvantage because it does not change the reaction mechanism qualitatively. To find the MEP to arbitrary accuracy, a steepest descent path can be traced from the saddle point(s) to the minima along the path (see Sec. VIII).

IV. OPTIMIZATION METHODS

The NEB method uses force projections [see Eq. (1)] to find the MEP. Optimization routines are responsible for moving the NEB along these forces to the MEP. The force pro-

jections place a limitation on the optimization methods used since the NEB forces ($\mathbf{F}^{NEB}$) are not conservative. Optimizers used with the NEB should not rely on the forces being consistent with an object function; rather, they should follow $\mathbf{F}^{NEB}$ until its magnitude drops below a specified criteria. Here, we consider five force-based optimizers: steepest descents (SD), 'quick-min' (QM),[1] fast inertial relaxation engine (FIRE),[12] conjugate gradients (CG),[13,14] and limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS).[15]

All of the optimizers were constrained with a maximum allowed step size for each NEB image. A value of 0.2 Å was chosen to prevent wild steps in high force regions while not significantly limiting performance in well-behaved low force regions. This parameter could, in principle, be thought of as an adjustable parameter and even used to control an unstable optimizer, but we did not find this to be a good strategy. For stable optimizers, performance was insensitive to the maximum step size, and for an unstable optimizer, controlling it with the maximum step size would result in an inefficient method resembling steepest descents. For this reason, we left the maximum step size fixed for all tests.

A. Steepest descents

The SD method follows the force vector from an initial configuration to a zero in the force. Given a configuration $\mathbf{R}_j$ at iteration j, the SD step moves to a new configuration

$$\mathbf{R}_{j+1} = \mathbf{R}_j + \alpha \mathbf{F}_j, \tag{5}$$

where $\mathbf{F}_j$ is the force and α is an adjustable parameter. If α is chosen to be the inverse of the curvature along the step direction, the optimizer will step directly to the minimum along the $\mathbf{F}_j$ direction. To ensure stability of the method, α should be less than $1/k_{max}$, where k_{max} is the maximum curvature in the system. The SD method is known to converge slowly in stiff systems.[14] Here, it is presented as the simplest optimizer to which better methods can be compared.

B. Quick-min

The QM optimizer improves upon the SD method by accelerating the system in the direction of the force, making the minimization more aggressive.[1] QM is a damped dynamics routine, where the damping parameter is replaced by a projection of the velocity along the force. The QM method can be coupled with a velocity Verlet algorithm,[16] or as described here, with an Euler integrator.

(1) Project the velocity in the direction of the force,

$$\mathbf{V}_j = (\mathbf{V}_j \cdot \hat{\mathbf{F}}_j) \hat{\mathbf{F}}_j.$$

(2) Zero the velocity if it is antiparallel to the force,

if $\mathbf{V}_j \cdot \hat{\mathbf{F}}_j < 0$ then $\mathbf{V}_j = 0$.

(3) Take an Euler step,

$$\mathbf{R}_{j+1} = \mathbf{R}_j + \Delta t \mathbf{V}_j,$$

$$\mathbf{V}_{j+1} = \mathbf{V}_j + \Delta t \mathbf{F}_j.$$

C. Fast inertial relaxation engine

Like QM, the FIRE algorithm takes dynamical steps and resets the velocity if the force and velocity are in opposite directions. In addition, FIRE employs a variable time step algorithm. The main difference is that QM projects the velocity onto the force vector, whereas FIRE only projects a component of the velocity in the force direction, while maintaining momentum in other directions. Details of this method can be found in Ref. 12.

D. Conjugate gradients

The CG method improves upon the SD method by following conjugate search directions instead of always following the force. The algorithm employed is the Polak-Ribière formula.[13,14]

(1) Initialize the search direction along the force,

$$\mathbf{d}_0 = \mathbf{F}_0.$$

(2) Calculate the step size λ using a line minimizer,

$$\mathbf{R}_{j+1} = \mathbf{R}_j + \lambda \mathbf{d}_j.$$

(3) Evaluate the new conjugate search direction,

$$\mathbf{d}_{j+1} = \mathbf{F}_{j+1} + \gamma \mathbf{d}_j,$$

where $\gamma = \mathbf{F}_{j+1} \cdot (\mathbf{F}_{j+1} - \mathbf{F}_j) / |\mathbf{F}_j|^2$. In our implementation, a single Newton's method step was used to minimize the force along the search direction $\mathbf{d}_i$ at each iteration. The derivative of the force along the search direction was evaluated with a finite difference step so that each CG iteration requires two force evaluations.

E. Limited-memory Broyden-Fletcher-Goldfarb-Shanno

The L-BFGS method is a quasi-Newton method that builds up information about the second derivatives during optimization and uses this information to step toward the predicted harmonic minimum.[14,15] Specifically, the inverse Hessian matrix $\mathbf{H}^{-1}$ is constructed iteratively, starting from a diagonal matrix. The L-BFGS method can be used in two ways. First, similar to CG, a search direction,

$$\mathbf{d}_j = \mathbf{F}_j \mathbf{H}_j^{-1}, \tag{6}$$

is identified at each iteration, and a line minimizer is used to step along that direction,

$$\mathbf{R}_{j+1} = \mathbf{R}_j + \lambda \mathbf{d}_j. \tag{7}$$

This method is referred to as L-BFGS(line).

A second approach is to use $\mathbf{H}^{-1}$ directly to calculate the step,

$$\mathbf{R}_{j+1} = \mathbf{R}_j + \mathbf{F}_j \mathbf{H}_j^{-1}. \tag{8}$$

This method, which we call L-BFGS(Hess), requires only one force call per iteration [instead of two for L-BFGS(line)] because there is no finite difference step to calculate λ. However, L-BFGS(Hess) requires a more conservative value for the diagonal elements of the initial inverse Hessian. Values that are too large result in oscillatory or wild behavior, and

values that are too small lead to slow optimization. Both flavors of the L-BFGS algorithm are tested here.

The L-BFGS method uses a memory of previous iterations to build the inverse Hessian. The number of iterations in the memory is a variable parameter. In all cases, we have used a value of 25 and found negligible changes in performance as long as this value is set large enough (greater than 10); however, a larger value for the memory stabilizes the optimizer for a poor initial $\mathbf{H}^{-1}$.

F. Global L-BFGS

We also tested a global version of L-BFGS (GL-BFGS) in which all images along the band are minimized with a single instance of the optimizer, instead of having an optimizer for each image.[4,17,18] In both the local and global L-BFGS methods, the images are optimized collectively, but in the GL-BFGS method, the interimage interactions are included in the inverse Hessian. The dimensionality of this matrix is $3 \times N \times P$ in GL-BFGS, where 3 is the dimensionality of space, N is the number of atoms, and P is the number of images along the band. In the local (image-by-image) L-BFGS method, there are P matrices, each with dimension of $3 \times N$. In the GL-BFGS method, the configuration of the band and the force acting on it are described by the vectors,

$$\mathbb{R} = (\mathbf{R}_1, \mathbf{R}_2, \ldots, \mathbf{R}_P),$$

$$\mathbb{F} = (\mathbf{F}_1, \mathbf{F}_2, \ldots, \mathbf{F}_P). \tag{9}$$

The optimizer then works the same as the L-BFGS using $\mathbf{H}^{-1}$ (for the entire band) to either directly calculate the optimization step in GL-BFGS(Hess) or with a line minimizer to calculate the step size (λ) in GL-BFGS(line).

V. NEB CONVERGENCE

The convergence of the NEB with the different optimizers was tested using a model system (see Ref. 19 for details) of the rearrangement mechanisms of a heptamer island diffusion on a fcc(111) crystal surface. The surface was formed from six layers of a fcc crystal, with 56 atoms per layer and the three bottom layers held frozen in their equilibrium bulk positions. A seven atom heptagonal island was placed on the surface with each atom in hollow sites. Atomic interactions were described by a pairwise Morse potential,

$$V(r) = D_e[e^{-2\alpha(r-r_0)} - e^{-\alpha(r-r_0)}], \tag{10}$$

with parameters fit to Pt,[20] $D_e = 0.7102$ eV, $\alpha = 1.6047$ Å^{-1}, and $r_0 = 2.8970$ Å. The potential was cut and shifted to zero at a distance of $r = 9.5$ Å.

Figure 2 shows the most stable initial state for the island and the thirteen product states with the lowest diffusion barriers. The NEB optimizers were tested by forming a linear band from the initial state to each final state and then counting the number of potential evaluations (force calls) required to reach convergence. Convergence was reached when the magnitude of the force on all images was less than a specified maximum force $\mathbf{F}_{max}$, so that
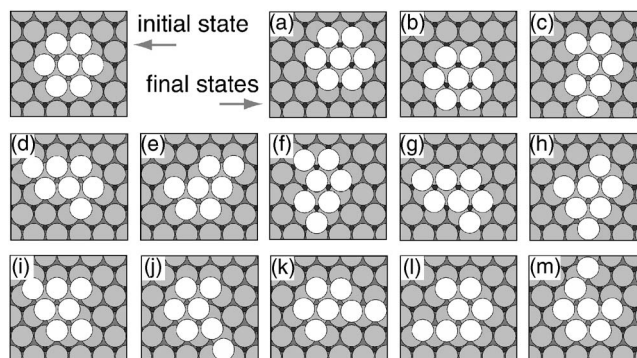

FIG. 2. Low energy rearrangement processes for a heptamer island on a (111) surface. A pairwise Morse potential is used with parameters adjusted to model Pt.

$$[\mathbf{F}_i^{NEB} \cdot \mathbf{F}_i^{NEB}]^{1/2} < \mathbf{F}_{max} \tag{11}$$

for each image i.

A comparison of the efficiency of the different optimizers with an eight image CI-NEB is presented in Table I. The average numbers of force calls per image to reach force criteria, $\mathbf{F}_{max}$, of 0.01 and 0.001 eV/Å for the 13 processes illustrated in Fig. 2 are tabulated. Of the first-order methods (SD, QM, and FIRE), which do not explicitly calculate curvatures of the potential, SD is the least efficient. QM is better, but the recent modifications made in the FIRE algorithm result in a significantly more efficient algorithm for this system. The second order methods that use line optimizers [CG, L-BFGS(line), and GL-BFGS(line)] are slightly less efficient than FIRE. They require fewer iterations to converge, but because each iteration involves two force evaluations to evaluate the curvature along a line by finite difference, the overall efficiency is lower. The Hessian based second order L-BFGS(Hess) has the potential to be twice as fast as the line optimizer version, L-BFGS(line), but it is unstable when used with the NEB. The global version, GL-BFGS(Hess), is stable and is half the cost of the GL-BFGS(line) method. This algorithm is 35% faster than FIRE and the most efficient algorithm that we have tested for optimizing the NEB.

The instability of second order algorithms when used with the NEB has been noted previously,[21,22] and associated

TABLE I. Average number of force calls to converge an eight image CI-NEB for heptamer island rearrangement with different optimizers. The NEB was considered converged when the magnitude of the force on each image dropped below the specified maximum force, satisfying: $\max |\mathbf{F}_i^{NEB}| < \mathbf{F}_{max}$.

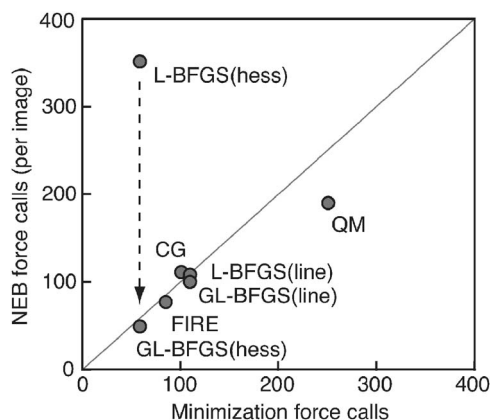| Optimization method	Force calls to reach $\mathbf{F}_{max}$ (eV/Å) of	
	0.01	0.001
SD: steepest descent	412	737
QM: quick-min	190	354
FIRE: fast inertial relaxation engine	77	116
CG: conjugate gradient	111	196
L-BFGS(line)	108	154
L-BFGS(Hess)	351	428
GL-BFGS(line): global L-BFGS(line)	100	147
GL-BFGS(Hess): global L-BFGS(Hess)	49	73

FIG. 3. Optimizer performance with the NEB as compared to minimization. Points on the line indicate that it takes the same average number of force calls to minimize from the midpoint of each heptamer rearrangement process as it does for an image in the NEB to converge to the MEP (with $F_{max}=0.01$ eV/Å). In general, the optimizers have a similar performance with the NEB as with minimization, and the second order methods (CG and L-BFGS) are better than the commonly used QM algorithm. The FIRE algorithm is also a significant improvement over QM. The Hessian based L-BFGS is twice as fast as the line-optimizer version but only when optimization is done globally for the NEB.

with the NEB lacking a simple Lagrangian (or object function). The F^{NEB} forces are constructed with projections [see Eq. (1)], resulting in nonconservative forces and a non-Hermitian Hessian matrix.[22] In this respect, optimizers that rely on curvature information could be inefficient or unstable when used with the NEB, but it is not well understood why some optimizers have stability problems and others do not.

To better understand the interaction of the optimizers and the NEB, we compared the performance of each for NEB optimization with geometry minimization. Minimization is based on conservative forces, so problems due to the nonconservative NEB forces should show up in the comparison. To make the minimization and NEB calculations as similar as possible, we minimized a geometry from the midpoint of each heptamer rearrangement process (see Fig. 2) and compared the average number of force calls required to reach $F_{max}=0.01$ eV/Å, with the number of force calls (per image) required to converge the NEB with the same force criteria.

Figure 3 shows that the optimizer performance is very similar for the NEB and minimization calculations, with the notable exception of the L-BFGS(Hess) optimizer. This optimizer works well for a single image but is unstable when used with the NEB. There is, however, a simple solution in the GL-BFGS(Hess) optimizer. The use of a global inverse Hessian for the entire NEB, in which interimage interactions are included, stabilizes the method and results in our most efficient optimizer.

The improved efficiency of GL-BFGS over the L-BFGS optimizer can be understood by looking at the Hessian matrices that each method constructs and comparing them to Newton's method which uses the true Hessian of the system. To make this comparison, we use a simple system in which eight colinear images are connected by springs of strength $k=5$ eV/Å^2. The exact global Hessian for this system, shown in Fig. 4, has values of $2k$ on the diagonal and $-k$ on
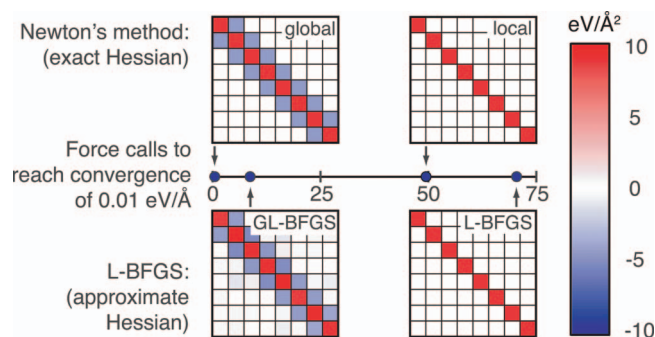

FIG. 4. (Color) Number of iterations required to converge a string of images connected by springs on a line. The L-BFGS and GL-BFGS optimizers are compared to Newton's method, using Hessian matrices from both a local (image-by-image) and global (entire band). In this harmonic system, Newton's method with the global Hessian converges in one iteration, and GL-BFGS learns the full Hessian and converges in eight iterations. When these optimizers are restricted to local (diagonal) Hessian matricies, both are inefficient. This indicates that the GL-BFGS optimizer is more efficient than the L-BFGS method because it learns information about the interimage (off-diagonal) elements of the Hessian.

the off-diagonal elements. This is a harmonic system, so this exact Hessian is independent of geometry, and Newton's method converges in a single iteration. Newton's method can also be applied locally (image by image). Here, the Hessian for each image is the single value of $2k$, and the local Newton's method step is made by using a global Hessian solely with this value on the diagonal. Since this is not the exact Hessian for the full system, Newton's method takes longer (48 iterations) to converge.

The same comparison is made for L-BFGS and GL-BFGS. The L-BFGS is a local method (image by image), so it learns only the diagonal elements of the Hessian. It takes several iterations to learn this Hessian and converges in 69 iterations—similar to the local implementation of Newton's method. The GL-BFGS optimizer is able to approximate the global Hessian, which is nearly identical to the exact Hessian by the time it converges in eight iterations (see Fig. 4). The off-diagonal elements of the Hessian are important for the global optimization of the band, and when they are included, the optimizer is significantly more efficient.

Beads on a line connected by springs is a simplified problem because the force projections of a NEB in higher dimensions are not present. Figure 5 shows a comparison of the final global Hessian for an eight image NEB on the two-dimensional London-Eyring-Polanyi-Sato (LEPS) potential coupled to a harmonic oscillator[1] (see Fig. 9). The true Hessian, calculated at the converged NEB, is not Hermitian since the forces are not conservative. One can see, however, the strong curvatures due to the potential on the block diagonal. The spring and tangent interactions between images appear in the off-diagonal blocks. The climbing image can be seen as the fifth image. Here, there are no off-diagonal elements since displacements of the neighboring images do not induce forces on the climbing image. This is why the climbing image tends to converge faster than other images when local optimizers are used. The GL-BFGS optimizer is forced to build up a symmetric Hessian. The (large) diagonal elements are similar to the true Hessian, and interimage interactions are present, but the final Hessian differs significantly from
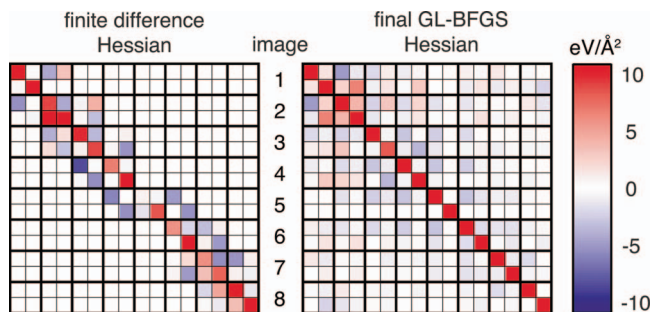
FIG. 5. (Color) The exact (finite difference) Hessian for an eight image converged NEB on the two-dimensional LEPS potential coupled to a harmonic oscillator, compared with the approximate Hessian constructed during a GL-BFGS optimization. The exact Hessian shows the curvatures due to the potential on the block-diagonal elements and the interimage spring and tangent interactions on the off diagonal. The non-Hermitian nature of the NEB forces is apparent from the asymmetry in this matrix. The approximate GL-BFGS Hessian is necessarily symmetric, and only picks up some of the tridiagonal nature of the exact Hessian. These off-diagonal elements are quite important however; GL-BFGS converges 2.5 times faster than L-BFGS which only has the block-diagonal elements.

the finite difference Hessian. Clearly these interimage elements are important, since GL-BFGS converges the NEB 2.5 times faster than the L-BFGS optimizer, but the qualitative error in the GL-BFGS Hessian suggests that the method could be further improved.

Finally, it should be noted that the line-optimizer based methods [CG and L-BFGS(line)] do not suffer from the same instability seen in L-BFGS(Hess). In these methods, the line is chosen based on local information, but the curvature along the line is found by finite difference—globally for the entire NEB. This curvature is more accurate than predicted by the local (image-by-image) L-BFGS Hessian and results in stable methods, although at roughly twice the cost of GL-BFGS(Hess).

The fact that the stability of the L-BFGS optimizer is sensitive to how it is implemented could explain the conflicting reports about whether the method is unstable[21,22] or not[4,18] when used with the NEB and string methods. This later work, from the Wales group, used a global version of the L-BFGS method as implemented in their OPTIM code[17] and found it to be highly efficient for NEB calculations—which is consistent with what we find here.
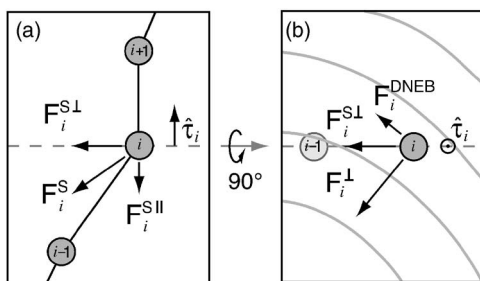

FIG. 6. The doubly nudged elastic band includes an additional force, $\mathbf{F}_i^{\text{DNEB}}$, to keep the band straight during convergence. This force is a component of the perpendicular spring force $\mathbf{F}_i^{S\perp}$, as shown in (A). The view in (B) is in the plane perpendicular to the band. Here, $\mathbf{F}_i^{\text{DNEB}}$ is shown as the component of $\mathbf{F}_i^{S\perp}$ which is orthogonal to the perpendicular force due to the potential $\mathbf{F}_i^{\perp}$.

VI. DOUBLY NUDGED ELASTIC BAND

The NEB force projection restricts the potential forces to act perpendicular to the band, and spring forces along the band. Since these forces are orthogonal, there is no competition between the potential forces pointing images to the MEP and the spring forces keeping them equally spaced. Also, since the spring forces act only along the band, there is no tendency for the springs to shorten the band; the spring forces on each image go to zero when they are equally spaced.

Trygubenko and Wales[4] proposed a double nudging modification to this projection scheme in which a component of the spring force acts perpendicular to the path. This component acts to straighten the band, keeping it shorter during convergence. Double nudging is very similar in spirit to the angular switching function introduced in the original NEB method[1] which was made unnecessary with an upwinding tangent.[2] The strength of the double nudging method is that it uses only the component of the perpendicular spring force which is not along the potential force, so that it does not cause corner cutting.[1] Figure 6 illustrates this projection and the resulting double nudging force $\mathbf{F}_i^{\text{DNEB}}$, which is added to the ith image in the NEB.

To write the double nudging force explicitly, we take the component of the spring force,

$$\mathbf{F}_i^S = k[(\mathbf{R}_{i+1} - \mathbf{R}_i) - (\mathbf{R}_i - \mathbf{R}_{i-1})], \qquad (12)$$

which is perpendicular to the tangent $\hat{\boldsymbol{\tau}}_i$,

$$\mathbf{F}_i^{S\perp} = \mathbf{F}_i^S - \mathbf{F}_i^S \cdot \hat{\boldsymbol{\tau}}_i \hat{\boldsymbol{\tau}}_i. \qquad (13)$$

Figure 6(a) illustrates the perpendicular and parallel components of the spring force, and Fig. 6(b) shows the plane normal to the tangent $\hat{\boldsymbol{\tau}}_i$ at image i. The perpendicular potential force $\mathbf{F}_i^{\perp}$ and the perpendicular spring force $\mathbf{F}_i^{S\perp}$ are both in this plane. The double nudging force $\mathbf{F}_i^{\text{DNEB}}$ is the component of $\mathbf{F}_i^{S\perp}$ which is orthogonal to $\mathbf{F}_i^{\perp}$,

$$\mathbf{F}_i^{\text{DNEB}} = \mathbf{F}_i^{S\perp} - \mathbf{F}_i^{S\perp} \cdot \hat{\mathbf{F}}_i^{\perp} \hat{\mathbf{F}}_i^{\perp}. \qquad (14)$$

The addition of this force to all (nonclimbing) images of the NEB is the DNEB.[4]

Trygubenko and Wales report that the addition of the double nudging force improves the stability of convergence with the NEB when using the L-BFGS optimizer.[4] They also mention that the DNEB cannot be used to accurately converge the MEP. The convergence problems are caused because the perpendicular spring force $\mathbf{F}_i^{S\perp}$ does not go to zero for a curved path and is only projected out if it is parallel to the potential force $\mathbf{F}_i^{\perp}$. Upon convergence, this perpendicular component of the potential force $\mathbf{F}_i^{\perp}$ goes to zero, so it will not project out the double nudging force, and the band will feel a straightening force. This frustration is illustrated in Fig. 7. The DNEB method can only converge the NEB to 0.2 eV/Å before it gets knocked away from MEP by the double nudging force.

The DNEB method was shown to work well when applied to long pathways with high initial forces,[4,18] which is not the conditions under which are testing it. To improve the convergence of the DNEB at lower forces, we introduced a
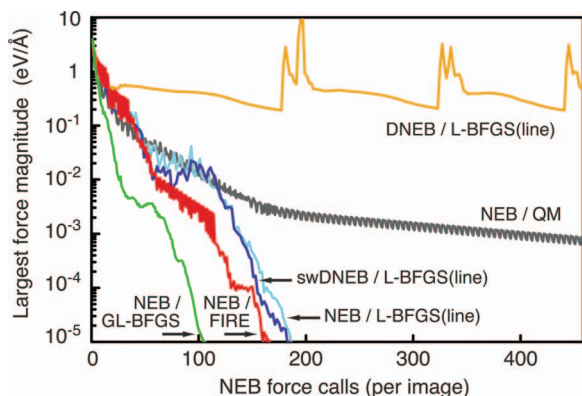
FIG. 7. (Color) Convergence of the NEB and DNEB with different optimizers, for the Pt island rearrangement process illustrated in Fig. 2(c). The DNEB has an instability at low force which prevents convergence. When the DNEB force is smoothly switched off during convergence (swDNEB), the performance is similar to the NEB. This plot also shows how the FIRE and GL-BFGS optimizers converge at a faster rate than QM.
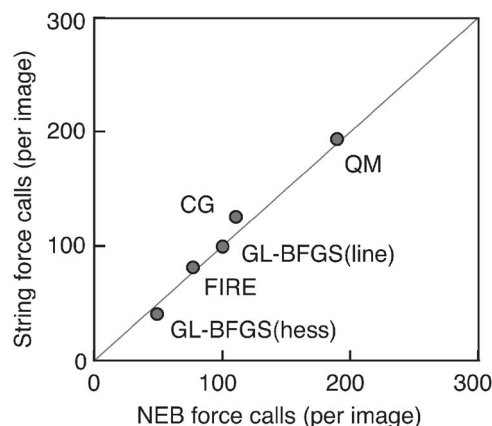

FIG. 8. Number of force calls required to converge the string and NEB methods to a force criteria of 0.01 eV/Å with different optimizers. The linear trend shows that the performance of the NEB and string methods (using the NEB upwinding tangent) are the same, and that GL-BFGS is the most efficient optimizer.

switching function which turns off $\mathbf{F}_i^{\mathrm{DNEB}}$ as the NEB converges and the magnitude of $\mathbf{F}_i^\perp$ drops below that of the spring force $\mathbf{F}_i^{S\perp}$,

$$\mathbf{F}_i^{\mathrm{swDNEB}} = \frac{2}{\pi}\tan^{-1}\left[\frac{|\mathbf{F}_i^\perp|^2}{|\mathbf{F}_i^{S\perp}|^2}\right]\mathbf{F}_i^{\mathrm{DNEB}}. \qquad (15)$$

Figure 7 shows the improved stability of this switched DNEB method (swDNEB) but also that it does not improve convergence over the regular NEB. It would be interesting to see if this method improves the stability of NEB calculations with high forces and also allows for accurate convergence, but such a test is beyond the scope of this work.

VII. STRING METHOD

The idea behind the string method is that a continuous reaction pathway (string) is optimized to the MEP.[5] Practically, however, the string method is very similar to the NEB in that the pathway is represented by a set of images connected by linear segments. The same tangent and force projections in the NEB (Ref. 2) are used to direct the images along the string to the MEP. A climbing image[3] can also be used to find the saddle point precisely. The one difference between the string and NEB methods is how the images are kept equally spaced along the pathway (or by some other specified distribution). In the NEB method, spring forces are introduced between images along the band, and the optimizer ensures equal spacing by minimizing these forces. In the string method, no spring forces are used. Instead, the images are kept equally spaced by repositioning them equally along the path after each iteration. Since the path is defined by the images, the repositioning step gives equal spacing to first order. Despite these different approaches for distributing images along the path, we find that there is no significant difference in the final path or the rate at which the methods converge. Figure 8 shows that it takes the same amount of work to converge the string and NEB methods with each optimizer. Either the NEB springs or the string redistribution can be used with the optimizers to efficiently find the MEP. These results are consistent with another recent study finding

that the NEB and string methods have similar performance when calculating isomerization pathways in small clusters and for folding pathways in peptides.[23] What is important for these methods is the definition of the tangent along the path, the force projections, and the climbing-image method for finding the saddle point.

VIII. SIMPLIFIED STRING METHOD

Recently, E *et al.* proposed a simplified version of the string method.[6] This method is described as being simpler than the NEB and string methods because it does not require a definition of the tangent along the path or the use of force projections. Instead, a cubic spline is used to parametrize the pathway between images. At each iteration, images are moved down the force and then redistributed along the spline. Convergence is based on how far the images move after each full iteration.

We have done two tests to compare the NEB and the simplified string method. First, we compare the accuracy of the methods, by calculating the root-mean-squared (rms) distance between the images on the converged bands and the true MEP. For this test, we used a LEPS potential coupled to a harmonic oscillator to form a two-dimensional potential energy surface.[1] Figure 9 shows this rms distance as a function of the number of images used in each method. We find two separate regimes; with more than 25 images, the string method more accurately reproduces the MEP, and with fewer, the string becomes unstable and the NEB is more accurate. The insets illustrate what is happening in these two regimes. Inset (B) shows how the string method follows the true MEP more closely than the NEB at a high-curvature region of the path. This is consistent with the findings of E *et al.* that the simplified string reproduces the MEP to a higher order than the NEB and will follow it more closely in the limit of many images.[6] In the 10–25 image range, the simplified string is poorly behaved and converged to a path that oscillates around the MEP [see inset (A)]. With fewer than 10 images, the simplified string oscillates without converging. We attribute this instability to the fact that the simplified
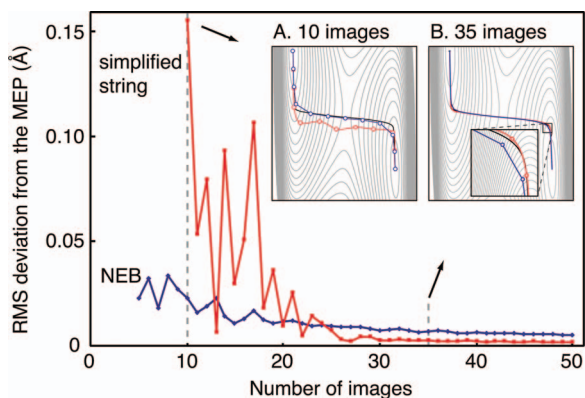
FIG. 9. (Color) Accuracy of the simplified string and NEB, measured by comparing the root-mean-squared (rms) deviation to the true MEP as a function of the number of images used to represent the path. For more than 25 images, the simplified string with a cubic spline interpolating function is closer to the MEP than the NEB with a one-sided upwinding tangent (B). Between 10 and 25 images, the simplified string develops oscillations and tends to deviate further from the MEP than the NEB method (A). With fewer than 10 images, the simplified string is unstable and does not converge. In this regime, the NEB remains stable, and is the preferred method.
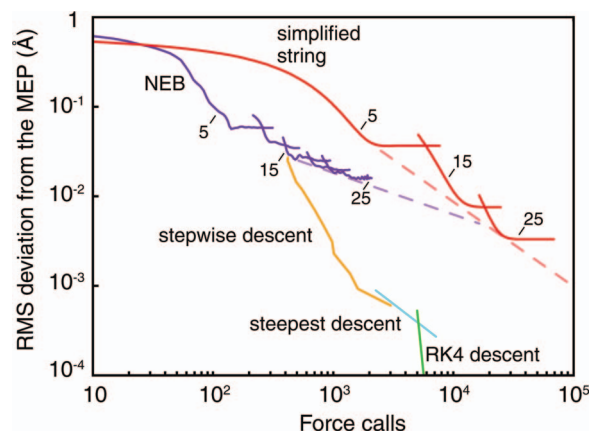

FIG. 10. (Color) The accuracy of MEP finding methods as a function of computational cost. The process chosen for this comparison is the Pt island rearrangement illustrated in Fig. 2(c). Accuracy is measured as the rms distance between the images of each method and the true MEP. Cost is measured by the total number of force evaluations. Two types of approaches are compared—the NEB and simplified string, chain-of-states methods, and methods which require finding the saddle first before minimizing down the MEP via a stepwise, steepest, or RK4 descent trajectory. For calculations with 25 images or less, the NEB is more efficient than the simplified string. For accurate calculations with more than 10^4 force calls, the simplified string becomes more efficient because it is based on the higher-order (RK4) optimizer and (cubic spline) interpolation between images. However, this is not as efficient as first finding the saddle (here with a five-image CI-NEB calculation) and then descending along the MEP. In this test, one should switch to a saddle-then-descend approach instead of using a 15 or greater image NEB.

string does not use an upwinding tangent. Tests (not presented here) show that the simplified string becomes unstable at the same point that a central-difference tangent becomes unstable, so we believe that the instability has the same root cause that is described in Ref. 2.

Using the rms distance to the MEP as a measure of accuracy illustrates a strength of the higher-order interpolation used in the simplified string method as compared to the NEB when the path is represented by many images. As discussed in Sec. III, however, this is not typically important for calculations of chemical rates. Taking Fig. 9(b) as an example—as long as both the simplified string and NEB include a climbing image to find the saddle, they describe the rate and mechanism of reaction equally well. Furthermore, once the saddle point is found, it is a simple calculation to descend from the saddle along the MEP.

The computational cost of calculating a MEP to arbitrary accuracy with the simplified string and NEB was compared for the Pt(111) island rearrangement process shown in Fig. 2(c). The accuracy of both methods is limited by the number of images used, so we performed tests with increasing numbers of images (5, 15, and 25) to compare the methods. The GL-BFGS optimizer was used for the NEB calculations and the fourth order Runga-Kutta (RK4) integrator with an optimized time step for the simplified string. RK4 is the suggested method for moving images down the force in the simplified string method;[6] the other optimizers discussed in this paper either cannot be used since the forces do not go to zero upon convergence, or they are not as efficient as RK4.

Figure 10 shows how the accuracy of the NEB and simplified string methods compare. The blue (NEB) and red (simplified string) curves labeled with the number 5 both have five images. The curve shows the accuracy of the images as they converge. Eventually, both curves plateau as the methods reach an intrinsic error due to the finite resolution of the bands. The higher accuracy of the simplified string can be seen by the smaller plateau value but also that it is

reached only after many more force calls. As more images are used, both methods become more accurate with an increasing cost. The dashed lines indicate the order of the methods—the NEB is of lower order since it uses a single sided tangent approximation, as compared to the cubic spline in the simplified string method; so eventually, it is more efficient to use the simplified string method. This is consistent with the findings of E et al.[6]

There is, however, a more efficient approach to find highly accurate MEPs. Once the saddle point(s) is found, the MEP can be followed by descending from the saddle(s) to the minima. The saddle point(s) can be found with arbitrary accuracy using a climbing-image NEB calculation with a minimal number of images, or using a min-mode following method.[10,11,24] Here, we used a five-image CI-NEB to find the saddle and then the dimer method to find the lowest mode at the saddle,[10] along which a descent trajectory is started. The cost of finding the saddle and the lowest mode was 203 force calls.

Several descent methods were used to trace out the MEP. A relatively inexpensive method, described here as stepwise descent, finds the next image that is a fixed distance down the MEP. The force on this next image is minimized using the quick-min optimizer until the force perpendicular to the path drops below a specified value. This method was repeated for increasingly higher path resolutions (with higher accuracy and cost) to trace out the orange line labeled "stepwise descent" in Fig. 10. At a cost of 3×10^3 force calls, the stepwise descent method becomes comparable to the steepest descent method which takes steps down the MEP in proportion to the force. Steepest descent is only stable for α below
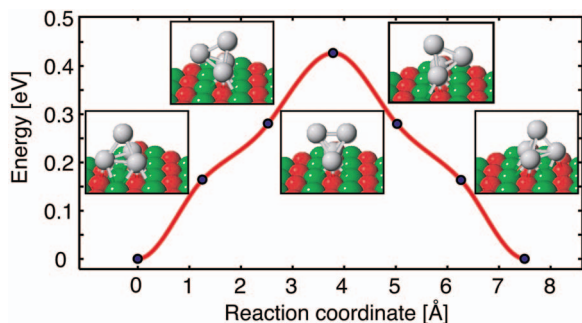
FIG. 11. (Color) A Pd_4 cluster diffuses by rolling on the MgO(100) surface. Gray atoms are Pd, red are O, and green are Mg.

FIG. 13. (Color online) Diffusion of O between hollow sites on Au(111).

0.04 eV/$Å^2$, so the stepwise descent is a good intermediate solution for tracing out the MEP for this potential. Finally, for even higher accuracy, the RK4 method is used to trace out the MEP. Here, a maximum time step of 0.55 fs could be used. Smaller time steps rapidly increased the accuracy of the MEP, making this the most efficient method for finding MEPs with an error smaller than 10^{-4} Å.

IX. DENSITY FUNCTIONAL THEORY CALCULATIONS

Computational efficiency is particularly important for finding reaction pathways when atomic interactions are based on first-principles calculations. The methods compared here require forces and energies, which are available from density functional theory (DFT) calculations. Here, we test our results of Sec. V on two systems with very different kinds of atomic interactions, described by DFT: the diffusion of a Pd tetramer on the MgO(100) oxide surface[25] and the dissociative adsorption and diffusion of oxygen on Au(111).

Our DFT calculations are performed with the Vienna *ab initio* simulation package, VASP. Electronic wavefunctions are described with a plane-wave basis set. Exchange and correlation are modeled with the PW91 generalized gradient approximation functional.[26] Ultrasoft pseudopotentials of the Vanderbilt form[27] are used to smoothen the wavefunctions within the core atomic regions, and these wavefunctions are orthogonalized to a frozen core within the projector augmented wave framework.[28] A plane-wave cutoff of 250 eV was set appropriate for the pseudopotentials. In the MgO oxide system, a single Γ-point calculation was sufficient for
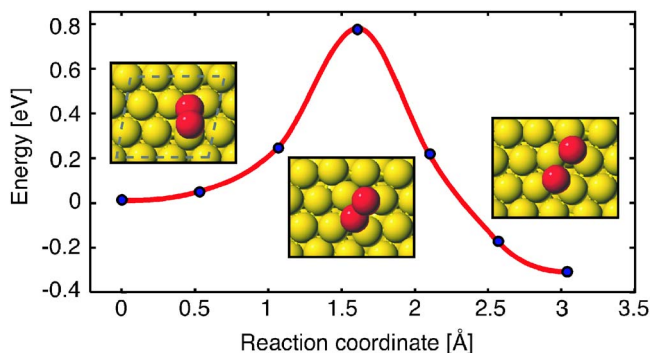
sampling the Brillouin zone, and for the Au surface, a 4 ×4×1 Monkhorst-Pack mesh[29] was used.

A. Pd_4 diffusion on MgO(100)

The MgO(100) surface is modeled with three layers, each with 36 atoms. The bottom two layers are frozen in bulk lattice sites and the top layer is relaxed to accommodate surface interactions. The diffusion process investigated is a Pd tetramer rolling over one edge to a mirrored final state (see Fig. 11). The optimizers show the same performance trend as in Sec. 5, with GL-BFGS outperforming the standard QM algorithm by a factor of 2. It should be noted here, as stated in Sec. III, that the optimizer parameters were not optimized specifically for the system; rather, default values were used. For QM and FIRE, a time step of 0.10 (in 10.18 fs units), and for GL-BFGS, the initial diagonal $\mathbf{H}^{-1}$ was scaled to have an inverse curvature of 0.05 $Å^2$/eV.

B. O_2 dissociation and diffusion on Au(111)

The dissociative adsorption of O_2 on Au(111) (Fig. 12) was calculated to test our results on a system with a strong covalent bond. The Au(111) surface was modeled as a four layer slab with nine atoms per layer and the top two layers relaxed. This reaction is qualitatively different from both the heptamer Pt island rearrangement and the Pd tetramer rollover processes because of large variance in bond strengths. To converge the NEB with GL-BFGS(Hess), a reduced scaled $\mathbf{H}^{-1}$ of 0.01 $Å^2$/eV was necessary. For stability, this must be set smaller than the inverse of the stiffest mode. The

FIG. 12. (Color online) Reaction mechanism of O_2 dissociative adsorption on Au(111) into adjacent fcc hollow sites.

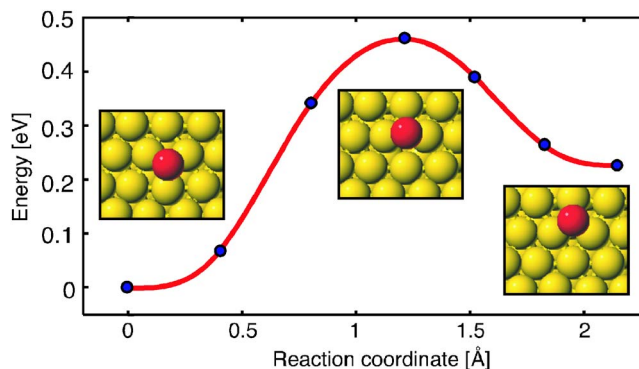TABLE II. Number of force calls required to converge a five-image CI-NEB for three different reaction mechanisms described by DFT. In each case, our GL-BFGS(Hess) optimizer outperformed existing methods, although this was particularly clear for the diffusion processes (I and III). Breaking of the stiff O–O bond in process II requires a conservative initial Hessian for the GL-BFGS method, which limited its convergence rate as compared to the damped dynamics methods. The NEB was considered converged when the magnitude of the force on each image dropped below 0.01 eV/Å.

Process	QM	FIRE	GL-BFGS
(I) Pd_4 diffusion on MgO(100)	178	148	98
(II) O_2 dissociation on Au(111)	187	147	126
(III) O diffusion on Au(111)	59	128	33

O–O stretch mode is calculated as 48.1 eV/Å^2, requiring an initial scaling of $\mathbf{H}^{-1}$ smaller than 0.02 Å^2/eV.

To verify that the lower GL-BFGS performance (only 15% faster than FIRE) is due to the stiff O–O mode and conservative $\mathbf{H}^{-1}$, we choose a second process involving O diffusion on the Au(111) slab (see Fig. 13). Here, the highest curvature is 9.4 eV/Å^2, which is small enough to use our more aggressive $\mathbf{H}^{-1}$ value of 0.05 Å^2/eV. As shown in Table II, the performance gap between GL-BFGS and QM/FIRE increases, similar to what we found for the Pd tetramer and Pt island rearrangement systems.

X. DISCUSSION

In our comparison of different methods, we have chosen a few test systems, a convergence criteria, and a limited number of optimizers. By making these choices, there is a danger of omitting methods, optimizers, or focusing on types of systems that do not emphasize the strengths and weaknesses of different approaches. Some of the methods that have not been compared here are the adaptive NEB,[21] the superlinear minimization scheme for the NEB,[22] the growing string,[30] the quadratic string method,[31,32] and a combination of double and single ended searches.[2,4]

Direct comparisons between methods is made easier when tests are made on easy-to-implement benchmark systems (such as the pairwise Morse potential used here) and when the code for algorithms is made available. To facilitate such comparisons in the future, the codes used for the calculations in this manuscript are available at Ref. 33.

XI. CONCLUSIONS

Our comparison of methods for finding minimum energy paths shows that when the same force projection and tangent definition are used, the method by which images are kept equally spaced (springs in the NEB method or redistribution with the string method) does not have a significant impact on computational efficiency; the optimizer is more important. Here, we show that a global implementation of the L-BFGS method, in which interimage curvatures are included in the memory of the optimizer, is the most efficient approach. Also, we show that the FIRE optimizer tends to be more efficient than the quick–min and is the preferred optimizer that does not (explicitly) rely on curvature information. Both the NEB and string methods can incorporate a climbing image to efficiently find a saddle point while using as few images as possible to represent the MEP. To find a high accuracy MEP, it is more efficient to first find the saddle and then descend to minima, as compared to using a chain-of-states method (NEB, string, or simplified string) with many images. All methods compared are force based and can be used with DFT calculations. The most efficient methods have been shown to work well both for empirical pairwise potentials and for metal-oxide and covalent bond breaking reactions at surfaces as described by DFT.

[1] H. Jónsson, G. Mills, and K. W. Jacobsen, *Classical and Quantum Dynamics in Condensed Phase Simulations*, edited by B. J. Berne, G. Ciccotti, and D. F. Coker (World Scientific, Singapore, 1998), pp. 385–404.
[2] G. Henkelman and H. Jónsson, J. Chem. Phys. **113**, 9978 (2000).
[3] G. Henkelman, B. P. Uberuaga, and H. Jónsson, J. Chem. Phys. **113**, 9901 (2000).
[4] S. A. Trygubenko and D. J. Wales, J. Chem. Phys. **120**, 2082 (2004).
[5] W. E, W. Ren, and E. Vanden–Eijnden, Phys. Rev. B **66**, 052301 (2002).
[6] W. E, W. Ren, and E. Vanden–Eijnden, J. Chem. Phys. **126**, 164103 (2007).
[7] L. R. Pratt, J. Chem. Phys. **85**, 5045 (1986).
[8] R. Elber and M. Karplus, Chem. Phys. Lett. **139**, 375 (1987).
[9] L. Xie, H. Liu, and W. Yang, J. Chem. Phys. **120**, 8039 (2004).
[10] G. Henkelman and H. Jónsson, J. Chem. Phys. **111**, 7010 (1999).
[11] L. J. Munro and D. J. Wales, Phys. Rev. B **59**, 3969 (1999).
[12] E. Bitzek, P. Koskinen, F. Fähler, M. Moseler, and P. Gumbsch, Phys. Rev. Lett. **97**, 170201 (2006).
[13] M. R. Hestenes and E. Steifel, J. Res. Natl. Bur. Stand. **49**, 409 (1952).
[14] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C: The Art of Scientific Computation*, 2nd ed. (Cambridge University Press, Cambridge, 1992).
[15] J. Nocedal, Math. Comput. **35**, 773 (1980).
[16] H. C. Andersen, J. Chem. Phys. **72**, 2384 (1980).
[17] D. J. Wales, *OPTIM: A Program for Optimizing Geometries and Calculating Reaction Pathways* (University of Cambridge Press, Cambridge, 2003) (http://www-wales.ch.cam.ac.uk/software.html).
[18] J. M. Carr, S. A. Trygubenko, and D. J. Wales, J. Chem. Phys. **122**, 234903 (2005).
[19] G. Henkelman, G. Jóhannesson, and H. Jónsson, *Progress on Theoretical Chemistry and Physics*, edited by S. Schwartz (Kluwer Academic, New York, 2000), pp. 269–299.
[20] D. W. Bassett and P. R. Webber, Surf. Sci. **70**, 520 (1978).
[21] P. Maragakis, S. A. Andreev, Y. Brumer, D. R. Reichman, and E. Kaxiras, J. Chem. Phys. **117**, 4651 (2002).
[22] J.-W. Chu, B. L. Trout, and B. R. Brooks, J. Chem. Phys. **119**, 12708 (2003).
[23] E. F. Koslover and D. J. Wales, J. Chem. Phys. **127**, 134102 (2007).
[24] R. Malek and N. Mousseau, Phys. Rev. E **62**, 7723 (2000).
[25] L. Xu, G. Henkelman, C. T. Campbell, and H. Jónsson, Phys. Rev. Lett. **95**, 146103 (2005).
[26] J. P. Perdew, *Electronic Structure of Solids*, edited by P. Ziesche and H. Eschrig (Akademie Verlag, Berlin, 1991), p. 11.
[27] D. Vanderbilt, Phys. Rev. B **41**, 7892 (1990).
[28] G. Kresse and J. Joubert, Phys. Rev. B **59**, 1758 (1999).
[29] H. J. Monkhorst and J. D. Pack, Phys. Rev. B **13**, 5188 (1976).
[30] B. Peters, A. Heyden, A. T. Bell, and A. Chakraborty, J. Chem. Phys. **120**, 7877 (2004).
[31] G. A. Cisneros, H. Liu, Z. Lu, and W. Yang, J. Chem. Phys. **122**, 114502 (2005).
[32] S. K. Burger and W. Yang, J. Chem. Phys. **124**, 054109 (2006).
[33] See: http://theory.cm.utexas.edu/henkelman/code/