Aalto University

OPEN ACCESS

This is an electronic reprint of the original article.
This reprint may differ from the original in pagination and typographic detail.

Koistinen, Olli-Pekka; Ásgeirsson, Vilhjálmur; Vehtari, Aki; Jonsson, Hannes

Nudged elastic band calculations accelerated with Gaussian process regression based on inverse interatomic distances

Published in:
Journal of Chemical Theory and Computation

DOI:
10.1021/acs.jctc.9b00692

Published: 10/12/2019

Document Version
Peer reviewed version

Please cite the original version:
Koistinen, O-P., Ásgeirsson, V., Vehtari, A., & Jonsson, H. (2019). Nudged elastic band calculations accelerated with Gaussian process regression based on inverse interatomic distances. *Journal of Chemical Theory and Computation*, *15*(12), 6738-6751. https://doi.org/10.1021/acs.jctc.9b00692

This material is protected by copyright and other intellectual property rights, and duplication or sale of all or part of any of the repository collections is not permitted, except that material may be duplicated by you for your research use or educational purposes in electronic or print form. You must obtain permission for any other use. Electronic or print copies may not be offered, whether for sale or otherwise to anyone who is not an authorised user.

Nudged elastic band calculations accelerated with Gaussian process regression based on inverse interatomic distances

Olli-Pekka Koistinen [†,‡,§] Vilhjálmur Ásgeirsson [‡] Aki Vehtari [†]
Hannes Jónsson [‡,§]

[†] Department of Computer Science, Aalto University, Espoo, Finland

[‡] Science Institute and Faculty of Physical Sciences, University of Iceland, Reykjavík, Iceland

[§] Department of Applied Physics, Aalto University, Espoo, Finland

hj@hi.is

Abstract

Calculations of minimum energy paths for atomic rearrangements using the nudged elastic band method can be accelerated with Gaussian process regression to reduce the number of energy and atomic force evaluations needed for convergence. Problems can arise, however, when configurations with large forces due to short distance between atoms are included in the data set. Here, a significant improvement to the Gaussian process regression approach is obtained by basing the difference measure between two atomic configurations in the covariance function on the inverted interatomic distances and by adding a new early stopping criterion for the path relaxation phase. This greatly improves the performance of the method in two applications where the original formulation does not work well: a dissociative adsorption of an H_2 molecule on a Cu(110) surface and a diffusion hop of an H_2O molecule on an ice Ih(0001) surface. Also, the revised method works better in the previously analyzed benchmark application to rearrangement transitions of a heptamer island on a surface, requiring fewer energy and force evaluations for convergence to the minimum energy path.

1 Introduction

Transitions involving rearrangements of atoms, such as chemical reactions or diffusion events, can be studied by analyzing a potential energy surface defined in a high-dimensional space of atom coordinates. Local minima on the energy surface represent stable states of the system, and minimum energy paths (MEP) connecting those characterize mechanisms of possible transitions. A maximum on an MEP corresponds to a first order saddle point on the energy surface, and the highest maximum provides an estimate of the activation energy for the transition.

An MEP can be defined from the requirement that any point on the path is at an energy minimum in all directions perpendicular to the path. A common way to find MEPs is the nudged elastic band (NEB) method,[1,2] where a discrete chain of atomic configurations, referred to as images, initially located along some trial path connecting the given minima, is iteratively

moved toward the nearest MEP. A typical NEB calculation requires on the order of a hundred evaluations of the energy and atomic forces (corresponding to the negative gradient vector of the energy) for each image, so the computational effort can be large, especially when it is combined with electronic structure calculations such as quantum wave function or density functional theory based methods. In addition, the calculation may need to be repeated if there are several possible final states for the transition. Thus, it is important to find ways to accelerate NEB calculations.

Peterson[3] applied machine learning based on neural networks[4,5] to accelerate NEB calculations by constructing an approximate energy surface for which the NEB calculations are carried out. After relaxation of the path on the approximate energy surface, the true energy and force are evaluated at the locations of the images of the relaxed path to see whether or not the path has converged to an MEP on the true energy surface. If not, the results of the new energy and force calculations are added to the training data set and the model is updated. This procedure is repeated iteratively until the approximate energy surface is accurate enough for convergence on the true MEP.

The GP-NEB method[6,7] applies a similar idea but uses Gaussian process regression (GPR)[8–11] to model the energy surface. As a non-parametric approach, GPR avoids difficulties related to optimization of a large number of parameters, which may cause problems when using, e.g., neural networks. Since NEB calculations are largely based on the atomic forces, a straightforward inclusion of derivative observations and prediction of derivatives can be seen as advantages of GPR for this application. As a probabilistic model, GPR also provides an uncertainty estimate, which can be used to further enhance the procedure by evaluating the energy and forces only at images located in the most uncertain region of the approximate energy surface before relaxing the path again.[7]

Sophisticated methods such as Gaussian approximation potentials[12,13] have been developed to model the entire potential energy surface of atomic systems with Gaussian process regression. The total energy is typically approximated as a sum of contributions of local atomic environments defined by descriptors that take into account the type of atoms involved as well as translational, rotational, and permutational invariance in the atomic configurations. Such methods could be coupled with the GP-NEB method, but since MEP calculations concern only a small part of the potential energy surface, it is convenient to keep the representation simple with respect to the atom coordinates and independent of the types of atoms involved.

The GP-NEB method based on a simple squared exponential covariance function[6,7] has been shown to work well for a benchmark problem involving thirteen different rearrangement transitions of a heptamer island on a solid surface,[14,15] reducing the number of energy and force evaluations by an order of magnitude as compared with a regular NEB calculation. A similar approach has been successfully applied also to the diffusion of a Au atom on an Al(111) surface and the diffusion of a Pt adatom across two terraces of a stepped platinum surface.[16] In some systems, however, strong and quickly changing repulsive forces may cause problems for a covariance function of this sort where the characteristic length scale and magnitude are the same throughout the configuration space. In atomic systems, it is typical that the potential energy changes faster with respect to the atom coordinates when atoms are close to each other, and this needs to be taken into account when improving the formulation of the covariance function, as shown here.

In this article, we present improvements to the GP-NEB method, specifically a better difference measure between a pair of configurations in the covariance function. Instead of measuring the distance between the two configurations in the space of atom coordinates, the measure is based on differences in inverted interatomic distances within each of the two configurations. In addition, a new early stopping criterion, restricting relative changes in the interatomic distances, is introduced to prevent atoms from moving too close to each other during the NEB relaxation

phase. The effect of the improvements is illustrated using a system where an H_2 molecule dissociates on a Cu(110) surface. The improved method is also applied to H_2O diffusion on ice Ih(0001) surface, another example for which the original formulation does not perform well. In addition, we show that the new features improve the performance of the GP-NEB method also in the previously analyzed[7] heptamer island benchmark.

2 Methods

In this section, we first briefly review the nudged elastic band method for completeness. In the second subsection, we describe how Gaussian process regression is used to model energy surfaces in the GP-NEB method and define an improved difference measure for the covariance function. Finally, the GP-NEB method is reviewed and a new early stopping criterion introduced in the third subsection.

2.1 Nudged elastic band method

The nudged elastic band method is an iterative algorithm for finding a minimum energy path connecting two given local minima on a potential energy surface.[1,2] The system can consist of atoms that move from one location to another in the transition as well as atoms that remain fixed at the same position. The number of moving atoms is denoted by N_m. An MEP is correspondingly a continuous path in a $3N_\mathrm{m}$-dimensional coordinate space. In the NEB method, the path is represented as a discrete chain of points, and each point is referred to as an image of the system. Starting from some initial path connecting the two minima, the basic idea is to move the images downhill on the energy surface to converge on the MEP and at the same time control the distribution of the images along the path. For the selection of the initial path, the simplest option is to use a straight line interpolation between the minima, but better alternatives are the so-called image dependent pair potential (IDPP) method,[17] which interpolates as closely as possible the distances between neighboring atoms, or the geodesic approach recently introduced by Zhu et al.[18]

During one iteration, a so-called NEB force vector is calculated for each intermediate image, and the images are then simultaneously moved in directions based on those vectors. The NEB force is a resultant of two components. The first one is perpendicular to the path and moves the chain toward the adjacent MEP. It is given by the negative energy gradient after removing the component parallel to the tangent of the path at each image. The other component is added to control the distribution of the images along the path, an artificial spring force acting only in the direction of the path tangent. When the spring constant is chosen to be the same for all pairs of adjacent images, an even spacing of the images along the path is obtained. Since the path is represented in a discretized way, the path tangent at an image needs to be estimated based on the locations of the neighboring images. A well-behaved estimate is obtained by defining the tangent to be parallel with the line segment connecting the current image to the neighboring image of higher energy or, if both of the neighbors are either higher or lower in energy than the current image, using a weighted average of the two line segments.[19] The algorithm has reached convergence when the magnitude of the NEB force on each image is below a given threshold, T_MEP.

Since the ultimate goal is to find the point of highest energy along the MEP, it is useful to make one of the images of the discrete chain converge to this maximum point. This can be accomplished with the climbing image nudged elastic band (CI-NEB) method,[20] where the highest energy image is treated differently. Whereas the component of the negative gradient parallel to the path tangent is normally removed from the NEB force, it is instead included and reversed for the climbing image, so as to point in the direction of increased energy along the path.

The spring force is not applied to the climbing image. In order to keep the intervals reasonably similar on both sides of the climbing image, the regular NEB method can be conducted first (using some preliminary convergence threshold) so that the image selected as the climbing image is not too far from its final location. The rest of the MEP is mainly needed to ensure that the highest saddle point has been identified and to provide an estimate for the path tangent at the climbing image. It is, therefore, practical to apply a tighter convergence threshold T_{CI} ($< T_{\mathrm{MEP}}$) to the climbing image.

In the GP-NEB calculations presented here, the iterative optimization of the locations of the images is performed using the velocity projection optimization algorithm.[2] It is based on the velocity Verlet algorithm,[21] but the velocity vector is projected on the direction of the NEB force vector to allow the images to accelerate in that direction. If the projected velocity and the NEB force point in opposite directions, as judged by the inner product, the velocity is set to zero. In the regular NEB calculations, which are compared to the GP-NEB results, also a global L-BFGS optimizer[22,23] implemented in the EON software package[24] is tested, and the more efficient one of the two optimizers is used in the reference method. The spring constant for the NEB force and the time step for the velocity projection optimization algorithm are chosen so that they work best for the regular NEB method.

2.2 Gaussian process regression

A Gaussian process (GP) is a flexible probabilistic model for functions in a continuous domain.[8–11] It is defined by a mean function $m(\mathbf{x})$, which controls the global mean level of the process (often set to zero), and a covariance function $k(\mathbf{x},\mathbf{x}')$, which defines how the function values $f(\mathbf{x})$ and $f(\mathbf{x}')$ at any two input points depend on each other:

$$\mathrm{Cov}\big[f(\mathbf{x}), f(\mathbf{x}')\big] = k(\mathbf{x},\mathbf{x}'). \tag{1}$$

If the covariance is large, the function values are likely to be similar, and with zero covariance they are considered independent. The joint probability distribution of the function values $\mathbf{f} = [f(\mathbf{x}^{(1)}), f(\mathbf{x}^{(2)}), \ldots, f(\mathbf{x}^{(N)})]^{\mathsf{T}}$ at any finite set of input points $\mathbf{X} = [\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(N)}]^{\mathsf{T}}$ is a multivariate Gaussian distribution $p(\mathbf{f}) = \mathcal{N}(\mathbf{m}, K(\mathbf{X},\mathbf{X}))$, where $\mathbf{m} = [m(\mathbf{x}^{(1)}), m(\mathbf{x}^{(2)}), \ldots, m(\mathbf{x}^{(N)})]^{\mathsf{T}}$ and the notation $K(\mathbf{X},\mathbf{X}')$ represents a covariance matrix

$$K(\mathbf{X},\mathbf{X}') = \begin{bmatrix} k(\mathbf{x}^{(1)},\mathbf{x}'^{(1)}) & k(\mathbf{x}^{(1)},\mathbf{x}'^{(2)}) & \cdots & k(\mathbf{x}^{(1)},\mathbf{x}'^{(N)}) \\ k(\mathbf{x}^{(2)},\mathbf{x}'^{(1)}) & k(\mathbf{x}^{(2)},\mathbf{x}'^{(2)}) & \cdots & k(\mathbf{x}^{(2)},\mathbf{x}'^{(N)}) \\ \vdots & \vdots & \ddots & \vdots \\ k(\mathbf{x}^{(N)},\mathbf{x}'^{(1)}) & k(\mathbf{x}^{(N)},\mathbf{x}'^{(2)}) & \cdots & k(\mathbf{x}^{(N)},\mathbf{x}'^{(N)}) \end{bmatrix}.$$

Thus, a GP can be seen as an infinite-dimensional generalization of the multivariate Gaussian distribution, serving as a prior probability distribution for the unknown function f. After evaluating the function at some training data points, the probability model is updated and a posterior probability distribution can be calculated for the function value at any point.

In the present application, f represents the energy of the system and

$$\mathbf{x} = [x_{1,1}, x_{1,2}, x_{1,3}, x_{2,1}, x_{2,2}, x_{2,3}, \ldots, x_{N_{\mathrm{m}},1}, x_{N_{\mathrm{m}},2}, x_{N_{\mathrm{m}},3}]^{\mathsf{T}}$$

is a $3N_{\mathrm{m}}$-dimensional configuration vector including the Cartesian coordinates for moving atoms $1, 2, \ldots, N_{\mathrm{m}} \in A_{\mathrm{m}}$. Given a training data set including both the energy and its gradient for certain configurations, the mean of the posterior process of f provides an approximate energy surface, which is here referred to as the GP approximation.

2.2.1 Covariance function and difference measures

Through selection of the covariance function, prior assumptions about the properties of function f can be encoded into the GP model. In the original formulation of the GP-NEB method,[6,7] a common choice to favor smooth functions was made by using the squared exponential covariance function

$$k_x(\mathbf{x},\mathbf{x}') = \sigma_\mathrm{c}^2 + \sigma_\mathrm{m}^2 \exp\left(-\frac{1}{2}\mathcal{D}_x^2(\mathbf{x},\mathbf{x}')\right), \tag{2}$$

where the difference measure

$$\mathcal{D}_x(\mathbf{x},\mathbf{x}') = \sqrt{\sum_{i=1}^{N_\mathrm{m}}\sum_{d=1}^{3}\frac{(x_{i,d}-x'_{i,d})^2}{l^2}} \tag{3}$$

is a regular Euclidean distance between configuration vectors $\mathbf{x}$ and $\mathbf{x}'$ in the $3N_\mathrm{m}$-dimensional space of the atom coordinates. The hyperparameters $\boldsymbol{\theta}_x = \{l, \sigma_\mathrm{m}\}$ control the length scale and magnitude of the covariance function, respectively, and σ_c^2 is an additional constant term with a similar effect as integration over an unknown constant intercept term having a Gaussian prior distribution with variance σ_c^2.

This type of covariance function is referred to as being stationary in that the characteristic length scale and magnitude of the model stay the same throughout the coordinate space. As will be demonstrated in the Results section, this can be problematic when representing the energy of atomic configurations, because the energy tends to change faster with respect to the atom coordinates when atoms are close to each other (see, e.g., the energy curve in Figure 1).

One way to make a stationary covariance function more tolerant toward this kind of non-stationary effects is to loosen its smoothness assumptions. The squared exponential covariance function produces infinite times differentiable sample functions, which means that the underlying energy surface is assumed to be extremely smooth. In other words, the model tends to avoid abrupt changes not only in the energy and its gradient but also in the derivatives of all orders. The Matérn family of covariance functions[25] allows control of the smoothness properties by including an additional hyperparameter, ν. These functions have a convenient form when ν is a half-integer. For example a choice of $\nu = \frac{3}{2}$ leads to once differentiable sample functions, which means that the gradient of the underlying function is assumed to be continuous but abrupt changes in the second derivatives are allowed. When $\nu \to \infty$, Matérn covariance function converges to the squared exponential covariance function.

As shown in the Supporting Information (SI), Matérn covariance functions with once ($\nu = \frac{3}{2}$) or twice ($\nu = \frac{5}{2}$) differentiable sample functions can perform better in modeling chemical systems than the squared exponential covariance function. A similar observation has been made recently with $\nu = \frac{5}{2}$ in ref 26. However, neither the squared exponential nor the Matérn covariance functions give good performance if the training data set includes configurations where the atoms come close to each other and the force acting on the atoms is large. In order to resolve this problem, we replace difference measure $\mathcal{D}_x(\mathbf{x},\mathbf{x}')$ in the squared exponential covariance function with a modified difference measure $\mathcal{D}_{1/r}(\mathbf{x},\mathbf{x}')$ that stretches when atoms approach each other and thus makes the covariance function nonstationary with respect to atom coordinates.

The difference measure $\mathcal{D}_{1/r}(\mathbf{x},\mathbf{x}')$ between configurations $\mathbf{x}$ and $\mathbf{x}'$ is defined through the sum of squared differences in the inverted interatomic distances between all atoms in the system, weighted by length scales specific to each atom pair type:

$$\mathcal{D}_{1/r}(\mathbf{x},\mathbf{x}') = \sqrt{\sum_{i\in A_\mathrm{m}}\sum_{\substack{j\in A_\mathrm{m}, j>i \\ \bigvee \\ j\in A_\mathrm{f}}}\frac{\left(\frac{1}{r_{i,j}(\mathbf{x})}-\frac{1}{r_{i,j}(\mathbf{x}')}\right)^2}{l^2_{\phi(i,j)}}}, \tag{4}$$

where

$$r_{i,j}(\mathbf{x}) = \sqrt{\sum_{d=1}^{3}(x_{i,d} - x_{j,d})^2}$$

is the distance between atoms i and j, $\phi(i,j)$ is the atom pair type for pair (i,j), and $l_{\phi(i,j)}$ is the length scale for that pair type. If frozen atoms are present, i.e., atoms that do not move during the transition, then pairs of two frozen atoms can be omitted in the calculation of the difference measure. Thus, the outer summation only includes the set of moving atoms A_m. The inner summation includes the set of frozen atoms A_f and part of the moving atoms so that each atom pair occurs only once. After a little rearrangement,

$$\left|\frac{1}{r_{i,j}(\mathbf{x})} - \frac{1}{r_{i,j}(\mathbf{x}')}\right| = \frac{|r_{i,j}(\mathbf{x}) - r_{i,j}(\mathbf{x}')|}{r_{i,j}(\mathbf{x})r_{i,j}(\mathbf{x}')}, \tag{5}$$

it is easy to see that the inversion of the interatomic distances corresponds to scaling the difference between the interatomic distances with their product. Thus, the closer two atoms are to each other, the larger effect a displacement of these atoms toward or away from each other has on the difference measure. On the other hand, if two atoms are far apart, the effect of changes in the interatomic distance becomes negligible.

In practice, some of the frozen atoms may be so far from the moving atoms that they can be omitted from the difference measure. The evaluations of the covariance function can, therefore, be sped up by defining an activation distance for the frozen atoms. In the applications presented in this article, a frozen atom is activated when it is within a radius of 5 Å from any moving atom in any configuration encountered during the GP-NEB algorithm. Once a frozen atom is activated, it stays active from then on and is taken into account when calculating covariances. The distances from the moving atoms to inactive frozen atoms are checked in each iteration, and if new frozen atoms are activated, the GP model is updated.

Replacing difference measure $\mathcal{D}_x(\mathbf{x},\mathbf{x}')$ with $\mathcal{D}_{1/r}(\mathbf{x},\mathbf{x}')$ in the squared exponential covariance function leads to the following form:

$$k_{1/r}(\mathbf{x},\mathbf{x}') = \sigma_\mathrm{c}^2 + \sigma_\mathrm{m}^2 \exp\left(-\frac{1}{2}\sum_{i\in A_\mathrm{m}}\sum_{\substack{j\in A_\mathrm{m}, j>i \\ \vee \\ j\in A_\mathrm{f}}} \frac{\left(\frac{1}{r_{i,j}(\mathbf{x})} - \frac{1}{r_{i,j}(\mathbf{x}')}\right)^2}{l^2_{\phi(i,j)}}\right). \tag{6}$$

Since $k_{1/r}$ corresponds to a regular squared exponential covariance function in the space of inverse interatomic distances which are obtained as functions of the original coordinates, it is a valid covariance function in the original coordinate space. With this covariance function, the GP-NEB method works well also for systems where strong chemical bonding is involved, as discussed in the Results section. Dealing with atomic forces and efficient optimization of the hyperparameter values $\boldsymbol{\theta}_{1/r} = \{l_1, l_2, ..., l_{N_\phi}, \sigma_\mathrm{m}\}$, where N_ϕ is the number of active atom pair types, require differentiation of the covariance function with respect to the atom coordinates and the hyperparameters. Expressions for the required partial derivatives for both k_x and $k_{1/r}$ are given in the Appendix.

2.2.2 Regression

Consider a regression problem $y = f(\mathbf{x}) + \epsilon$, where ϵ is a Gaussian noise term with variance σ^2, and a training data set $\{\mathbf{X}, \mathbf{y}\}$, where $\mathbf{y} = [y^{(1)}, y^{(2)}, \ldots, y^{(N)}]^\mathsf{T}$ includes noisy output observations from N input points $\mathbf{X}$. When modeling function f as a Gaussian process with a prior mean function $m(\mathbf{x}) = 0$ and a prior covariance function $k(\mathbf{x},\mathbf{x}')$, the posterior predictive

distribution for the function value $f(\mathbf{x}^*)$ at a new point $\mathbf{x}^*$, conditional on the hyperparameters $\boldsymbol{\theta}$ of the covariance function, is a Gaussian distribution with mean

$$\mathrm{E}[f(\mathbf{x}^*) \,|\, \mathbf{y}, \mathbf{X}, \boldsymbol{\theta}] = K(\mathbf{x}^*, \mathbf{X})(K(\mathbf{X}, \mathbf{X}) + \boldsymbol{\Sigma})^{-1}\mathbf{y} \tag{7}$$

and variance

$$\mathrm{Var}[f(\mathbf{x}^*) \,|\, \mathbf{y}, \mathbf{X}, \boldsymbol{\theta}] = k(\mathbf{x}^*, \mathbf{x}^*) - K(\mathbf{x}^*, \mathbf{X})(K(\mathbf{X}, \mathbf{X}) + \boldsymbol{\Sigma})^{-1}K(\mathbf{X}, \mathbf{x}^*), \tag{8}$$

where $\boldsymbol{\Sigma} = \sigma^2\mathbf{I}_N$ is a noise covariance matrix with $\mathbf{I}_N$ denoting an identity matrix of size N. The corresponding prediction for the partial derivative of f with respect to coordinate $x^*_{i,d}$ is given by

$$\mathrm{E}\left[\left.\frac{\partial f(\mathbf{x}^*)}{\partial x^*_{i,d}} \right| \mathbf{y}, \mathbf{X}, \boldsymbol{\theta}\right] = \frac{\partial K(\mathbf{x}^*, \mathbf{X})}{\partial x^*_{i,d}}(K(\mathbf{X}, \mathbf{X}) + \boldsymbol{\Sigma})^{-1}\mathbf{y}, \tag{9}$$

where the elements of $\partial K(\mathbf{x}^*, \mathbf{X})/\partial x^*_{i,d}$ are obtained by differentiating the covariance function. Expressions for the partial derivatives of covariance functions k_x and $k_{1/r}$ are given in the Appendix.

The derivatives of the covariance function are needed also for including derivative information in Gaussian process regression.[27–30] When $\mathbf{y}$ is extended to include partial derivatives of f at the training data points with Gaussian noise variance σ^2_d, the training covariance matrix $K(\mathbf{X}, \mathbf{X})$ is extended correspondingly to include prior covariances between the partial derivatives and function values

$$\mathrm{Cov}\left[\frac{\partial f(\mathbf{x})}{\partial x_{i,d}}, f(\mathbf{x}')\right] = \frac{\partial}{\partial x_{i,d}}\mathrm{Cov}\big[f(\mathbf{x}), f(\mathbf{x}')\big] = \frac{\partial k(\mathbf{x}, \mathbf{x}')}{\partial x_{i,d}} \tag{10}$$

and the covariances between the derivatives

$$\mathrm{Cov}\left[\frac{\partial f(\mathbf{x})}{\partial x_{i_1,d_1}}, \frac{\partial f(\mathbf{x}')}{\partial x'_{i_2,d_2}}\right] = \frac{\partial^2}{\partial x_{i_1,d_1}\partial x'_{i_2,d_2}}\mathrm{Cov}\big[f(\mathbf{x}), f(\mathbf{x}')\big] = \frac{\partial^2 k(\mathbf{x}, \mathbf{x}')}{\partial x_{i_1,d_1}\partial x'_{i_2,d_2}}. \tag{11}$$

The vector $K(\mathbf{x}^*, \mathbf{X})$, required in prediction at a new point $\mathbf{x}^*$, is extended similarly to include covariances between the function values $f(\mathbf{x}^*)$ at the new point and the partial derivatives at the training data points. The extension of the noise covariance matrix $\boldsymbol{\Sigma}$ consists of the noise variances of both the energy and derivative observations on the diagonal. Notice that since the function values are in different units than the derivatives, the numerical value of σ^2_d is not generally comparable to σ^2.

The hyperparameter values $\boldsymbol{\theta}$ can be optimized by defining a prior probability distribution $p(\boldsymbol{\theta})$ and maximizing the marginal posterior probability density $p(\boldsymbol{\theta} \,|\, \mathbf{y}, \mathbf{X}) \propto p(\boldsymbol{\theta})p(\mathbf{y} \,|\, \mathbf{X}, \boldsymbol{\theta})$, where

$$p(\mathbf{y} \,|\, \mathbf{X}, \boldsymbol{\theta}) = |2\pi(K(\mathbf{X}, \mathbf{X}) + \boldsymbol{\Sigma})|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\mathbf{y}^\mathsf{T}(K(\mathbf{X}, \mathbf{X}) + \boldsymbol{\Sigma})^{-1}\mathbf{y}\right) \tag{12}$$

is the marginal likelihood of $\boldsymbol{\theta}$ in light of the given training data set $\{\mathbf{X}, \mathbf{y}\}$. To improve robustness of the hyperparameter optimization, we use here weakly informative priors based on the range of the training data. The prior distributions used for $\boldsymbol{\theta}_{1/r}$ are $p(\sigma_\mathrm{m}) = \mathcal{N}(0, (\Delta_\mathbf{y}/3)^2)$ and $p(l_\psi) = \mathcal{N}(0, (\Delta_\mathbf{X}/3)^2)$, where $\Delta_\mathbf{y}$ is the difference between the highest and lowest observed energy values and $\Delta_\mathbf{X}$ is the maximum difference between the observed data points based on difference measure $\mathcal{D}_{1/r}(\mathbf{x}, \mathbf{x}')$ with unit length scales. In practice, both the objective function and the hyperparameters are transformed to logarithmic scale for the optimization. The fixed value of the constant term σ^2_c is set to the square of the mean of the observed energy values.

In the applications of the GP-NEB method presented here, both energy and gradient observations are assumed to be noiseless, but small values for the noise variances, $\sigma^2 = 10^{-8}$ eV^2

and $\sigma_\mathrm{d}^2 = 10^{-8}$ eV^2/Å^2, are used to avoid numerical problems when inverting the training covariance matrix. The GPR calculations were implemented using the GPstuff toolbox,[31] and the hyperparameters of the covariance function were optimized using the scaled conjugate gradient algorithm[32] whenever the model was updated.

2.3 GP-NEB method

The GP-NEB method[6,7] is an algorithm that accelerates NEB calculation by modeling the potential energy surface as a Gaussian process, relaxing the path on the approximated surface, and refining the model after new evaluations have been performed. There are two variations of the method. In the simpler version, referred to as the all-images-evaluated (AIE) algorithm, energy and atomic forces are evaluated at all intermediate images of the path after each NEB relaxation phase. Here we focus, however, on the more efficient one-image-evaluated (OIE) version,[7] where the true energy and gradient are evaluated only for the image that is located in the most uncertain region according to the GP model.

2.3.1 OIE algorithm

The OIE algorithm[7] is started by constructing an initial GP model based on the initial data from the two end points of the path and evaluating the energy and force at the most uncertain intermediate image of the initial path. The selection is based on the variance of the posterior predictive distribution of energy at each image. The GP model is then updated based on the obtained information, and the whole NEB path is relaxed on the revised GP approximation. By default, each NEB relaxation phase is started from the same initial path and continued until the maximum magnitude of the approximated NEB forces has dropped below a threshold $T_\mathrm{MEP}^\mathrm{GP} = T_\mathrm{CI}/10$, where T_CI is the final convergence threshold for the accurate NEB force on the climbing image. Other options are also possible in order to decrease the number of steps required for the relaxation.[7] The relaxation is first conducted without climbing image mode until a preliminary convergence threshold $T_\mathrm{CIon}^\mathrm{GP}$ is reached and then continued from the preliminary evenly spaced path with climbing image mode turned on.

The final convergence of the algorithm is defined similarly as in the regular NEB method, based on final convergence thresholds T_MEP and T_CI for the magnitude of the accurate NEB forces. However, since all intermediate images are relaxed after each evaluation, the accurate NEB force can only be known for one image at a time. To enable confirmation of the final convergence of the whole path with accurate NEB forces, the following rules are applied based on the mixture of accurate and approximated NEB forces after each model update: If the maximum magnitude of the accurate/approximated NEB forces is above T_MEP, the NEB relaxation phase is executed normally and the image with the highest uncertainty is evaluated. Otherwise, the climbing image is evaluated without moving the path (if not already evaluated). If the maximum NEB force magnitude is below T_MEP but the accurate NEB force magnitude on the climbing image above T_CI, the path is relaxed and the climbing image re-evaluated. Finally, if the maximum magnitude of the accurate/approximated NEB forces is below T_MEP and the accurate NEB force magnitude on the climbing image is below T_CI, then more images are evaluated without moving the path, starting from the image with the highest uncertainty, until all images have been evaluated or some of the NEB forces is again above T_MEP.

When the motivation for finding an MEP is to estimate the transition rates using harmonic transition state theory, additional force evaluations in the neighborhood of the end points of the path are usually required to estimate Hessian matrices at the two minimum points. By performing these evaluations already before the MEP calculation, the additional data can be included in the initial data set for the GPR calculations.[7] In the applications presented in this article, the Hessian data consist of one data point per input coordinate, including both energy

and gradient evaluated at a location given by a displacement of 10^{-3} Å in the positive direction of the coordinate axis.

2.3.2 Early stopping rules

To prevent the path from moving too far into regions with no observed data, it is good to have an early stopping rule for the NEB relaxation phase. The stopping criterion defined in the original formulation of the GP-NEB method[7] is based on the distance to the nearest evaluated configuration according to the regular difference measure $\mathcal{D}_x$: For all images $\mathbf{x}_{\mathrm{im}}$ of the current path, there needs to exist an evaluated configuration $\mathbf{x}_{\mathrm{eval}}$ so that

$$\mathcal{D}_x(\mathbf{x}_{\mathrm{im}}, \mathbf{x}_{\mathrm{eval}}) < L_x^{\mathrm{es}}. \tag{13}$$

If this condition does not hold, then the last NEB iteration is rejected, the relaxation phase is stopped, and the image that triggered the early stopping rule is evaluated next. By default, L_x^{es} is set to one half of the length of the initial path.

This stopping criterion does not, however, prevent the path from moving to locations where atoms come close together. As illustrated in the Results section and the SI, large repulsive forces between atoms may cause problems for the GP model when using a stationary covariance function with the regular difference measure $\mathcal{D}_x$. The inverse-distance difference measure $\mathcal{D}_{1/r}$ stretches in the direction of the interatomic force when the atoms are closer to each other, which effectively smoothens the repulsive forces with respect to the difference measure and makes the modeling easier. However, to ensure that the new evaluations are made at sensible locations, it is still good to restrict too large relative changes of interatomic distances in the NEB relaxation phase by an additional early stopping criterion: For all images $\mathbf{x}_{\mathrm{im}}$ of the current path, there needs to exist an evaluated configuration $\mathbf{x}_{\mathrm{eval}}$ so that

$$\forall i \in \mathrm{A_m}\, \forall j \in \mathrm{A_m} \cup \mathrm{A_f} : \frac{2}{3} r_{i,j}(\mathbf{x}_{\mathrm{eval}}) < r_{i,j}(\mathbf{x}_{\mathrm{im}}) < \frac{3}{2} r_{i,j}(\mathbf{x}_{\mathrm{eval}}). \tag{14}$$

In other words, each evaluated data point is surrounded by an allowed neighborhood with a limit for the relative (logarithmic) changes in the interatomic distances, and the position of an image is required to be inside some of these allowed neighborhoods.

The formulation of this early stopping criterion relies on the assumption that a reduction of an interatomic distance to two thirds of the bond length does not lead to problems when using a covariance function with the inverse-distance difference measure $\mathcal{D}_{1/r}$. If there exists no evaluated data from the repulsive region with interatomic distance shorter than the bond length, the early stopping rule keeps the path safe, and if such data exists, the shape of the GP model should lead the path away from those regions. Besides avoiding unphysical configurations, another function of the new early stopping rule is to generally stabilize the development of the GP model by constraining the exploration into regions of large uncertainty. The limit in the relative change of the interatomic distances can also be seen as a trade-off between confirming stability of the algorithm and optimizing its efficiency with respect to the number of evaluations required for convergence. Based on our tests, the value of $\frac{2}{3}$ is a good general choice for all the systems studied here, although for example $\frac{1}{2}$ or $\frac{3}{4}$ would be applicable as well. Even though the inverse-distance difference measure $\mathcal{D}_{1/r}$ handles well also strong repulsive forces, it is possible that a more restrictive limit becomes beneficial in some other systems.

From the perspective of avoiding regions with large uncertainty, it could seem tempting to base the stopping criterion on the uncertainty estimate of the GP model, which is now based on the inverse-distance difference measure $\mathcal{D}_{1/r}$. In the beginning, however, there would be a potential risk that a falsely large length scale of one atom pair type compared to another would make differences in the corresponding interatomic distances negligible in the expression

of $\mathcal{D}_{1/r}$ and the uncertainties in these directions would be underestimated. Since our definition for the early stopping criterion is independent of the length scales of the difference measure, it would be unaffected by the false length scales and would instead help to safely correct them by forcing evaluations to be made before moving too far in these directions. Instead of logarithmic scale, it would still be possible to connect the lower and upper limit based on changes in the inverse interatomic distances, which would increase the upper limit from $\frac{3}{2}$ to 2, but we find the logarithmic scale more intuitive if the user wants to modify the sensitivity of the stopping rule.

If the displacements of the images during a single iteration of the NEB relaxation phase were unlimited, using the early stopping rules would involve a potential risk for a loop where the same or almost the same configuration with high atomic force keeps throwing the path away from the allowed region. Since the early stopping rules reject the last NEB iteration, the new evaluation would always be made at that same location and the allowed region would not be extended. For this reason, we set additional limitation rules for the step length of the NEB iterations during the relaxation phase to guarantee that an evaluated image cannot move away from the allowed region during a single NEB iteration. Notice that these limitation rules do not stop the NEB relaxation phase but only reduce the step length of the NEB iterations when necessary.

In respect of the new early stopping criterion (eq 14), the limitation rule for image $\mathbf{x}_{\mathrm{im}}$ is defined as follows: An individual atom $i \in \mathrm{A_m}$ cannot move more than 99% of

$$\min_{j \in \mathrm{A_f} \cup \mathrm{A_m} \setminus \{i\}} r_{i,j}(\mathbf{x}_{\mathrm{im}})/6,$$

where the minimum is taken over all interatomic distances from that atom to any other atom in $\mathbf{x}_{\mathrm{im}}$. If this limit is exceeded, the whole displacement vector (including all moving atoms) is shortened so that the displacement of atom i is at the limit. This limitation rule guarantees that the interatomic distances cannot decrease to two thirds during a single NEB iteration.

A corresponding limitation rule to accompany the original early stopping criterion (eq 13) is obtained by limiting the displacement vector to 99% of L_x^{es}. If this limit is exceeded, the displacement vector is simply shortened to the limit.

3 Results

In this section, we present results showing the success of the improved covariance function with the new early stopping criterion in GP-NEB calculations for two systems that are challenging for the original formulation of the GP-NEB method. The problems encountered when using a stationary squared exponential covariance function are illustrated in context of the first application example, where a hydrogen molecule dissociates on a Cu(110) surface. The revised method is shown to perform well also in a more complicated system where an H_2O molecule makes a diffusion hop on an ice Ih(0001) surface. In addition, we show that the performance of the GP-NEB method is improved also for a previously analyzed benchmark application involving rearrangements of a heptamer island on a surface.

3.1 Application to H_2 dissociation on Cu(110)

A system where a hydrogen molecule dissociates on a Cu(110) surface[1] is a good example to illustrate the benefit of replacing the regular difference measure $\mathcal{D}_x$ with the inverse-distance difference measure $\mathcal{D}_{1/r}$ in the covariance function when modeling the energy surface with a Gaussian process. The copper slab representing the (110) surface consists of 216 Cu atoms in six layers, and the potential energy function representing the "true" energy is obtained as described in ref 1 using the embedded-atom method (EAM).[33] We start by illustrating the challenges that

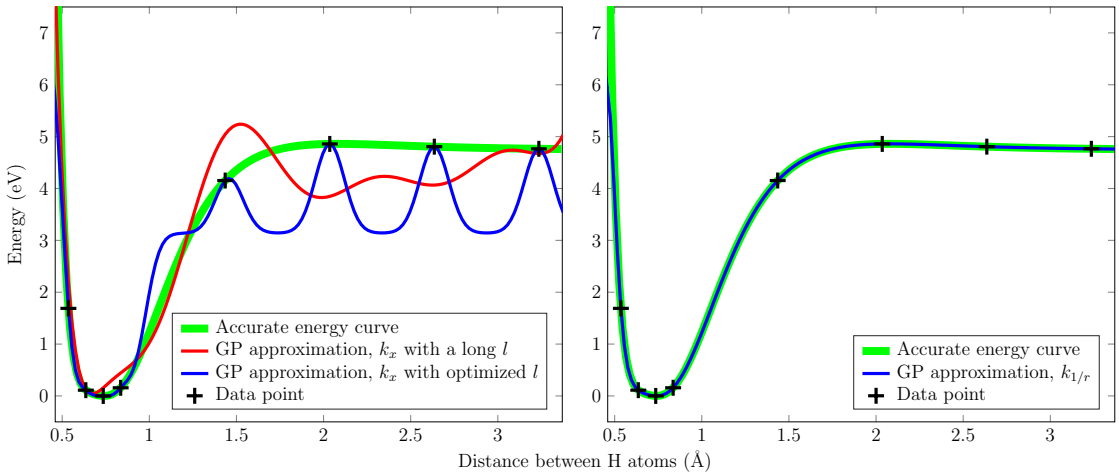

Figure 1: The thick green curve shows "true" energy as a function of distance between two hydrogen atoms. Training data for the GP models, marked with + signs, include accurate values for both energy and its first derivative with respect to the coordinate of the moving hydrogen atom. Left: GP approximations obtained using the stationary squared exponential covariance function k_x. The red curve shows a GP approximation obtained with a long length scale (fixed hyperparameters: $\sigma_\mathrm{m} = 1.6$ eV, $l = 1$ Å) and the blue curve with optimized hyperparameters ($\sigma_\mathrm{m} \approx 1.9$ eV, $l \approx 0.084$ Å). Right: GP approximation obtained using covariance function $k_{1/r}$, based on the inverse-distance difference measure $\mathcal{D}_{1/r}$, with optimized hyperparameters.

arise when modeling a one-dimensional energy curve for a two-atom system where a hydrogen atom approaches another hydrogen atom using the stationary squared exponential covariance function k_x, see Figure 1. The "true" energy is smoothly varying but rises sharply when the atoms are close to each other. If the length scale l in the covariance function is too long, the dominant data from the short distance region disturb prediction at longer distances. In the example shown by the red curve, the GP approximation does not go through the data points even if the assumed noise variance is set to be small. Consequently, the length scale tends to be optimized to a small value. With a short length scale, however, the GP model has problems in interpolating the flat region where atoms are farther away from each other, and the predicted values between the data points approach the mean of the data. When the regular difference measure $\mathcal{D}_x$ is replaced with the inverse-distance difference measure $\mathcal{D}_{1/r}$, the GP model manages to reproduce the energy curve without problems.

From the perspective of the GP-NEB algorithm, the oscillations in the GP approximation caused by a short length scale disturb the NEB relaxation phase since the path tends to move toward the fallacious energy minima. If the length scale is somewhat sensible, the oscillations should eventually disappear after additional energy and force evaluations, but the number of required evaluations may grow large especially in high-dimensional cases. In this case, however, data from a bit shorter distances would force the length scale to be so small that interpolation would become practically impossible.

Figure 2 shows a two-dimensional illustration of a cut through an energy surface for a hydrogen molecule dissociating on a Cu(110) surface. In spite of quite a dense grid of training data points, the GP model based on the regular difference measure $\mathcal{D}_x$ cannot recover from the oscillations caused by the high-gradient data on the left. And again, data points closer to the vertical axis would force the length scale to become even shorter and make things worse. With the inverse-distance covariance function $k_{1/r}$, the high-gradient data near the vertical axis do not cause problems for the GP model and the agreement with the "true" energy surface is again good.

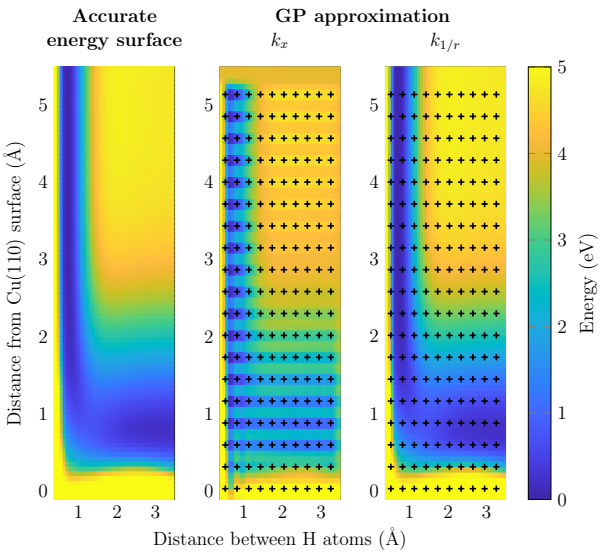

Figure 2: A two-dimensional cut through the potential energy surface for a pair of hydrogen atoms near a Cu(110) surface. The H–H molecular axis is parallel to the surface and perpendicular to the atom rows on the Cu(110) surface. The horizontal axis represents the distance between the two H atoms, and the vertical axis represents the distance between the H atoms and the Cu(110) surface. Left: 'True' energy, given by an energy surface taken from ref 1. Middle: GP approximation based on the grid of energy and atomic force evaluations shown with + signs when using the stationary squared exponential covariance function k_x with optimized hyperparameters. Notice the short length scale oscillations in the GP approximation. Right: GP approximation obtained using covariance function $k_{1/r}$, based on the inverse-distance difference measure $\mathcal{D}_{1/r}$, with optimized hyperparameters. In this case the GP approximation agrees well with the accurate energy surface.

GP-NEB calculations for finding the minimum energy path of H_2 dissociative adsorption on Cu(110) were performed with the improvements presented in the Methods section, including covariance function $k_{1/r}$ based on the inverse-distance difference measure $\mathcal{D}_{1/r}$ and the new early stopping criterion restricting relative changes of interatomic distances. The initial state represents an H_2 molecule far from the Cu(110) surface, while the final state represents two H adatoms sitting on the surface. Each NEB relaxation phase was started from an IDPP path with eight intermediate images, and the climbing image mode was turned on when the magnitude of the NEB force based on the GP approximation had dropped below $T_{\mathrm{CIon}}^{\mathrm{GP}} = 1$ eV/Å for all images. The GP-NEB algorithm was continued until the magnitude of the true NEB force had dropped below $T_{\mathrm{CI}} = 0.01$ eV/Å for the climbing image and below $T_{\mathrm{MEP}} = 0.3$ eV/Å for the other intermediate images. A spring constant of 1 eV/Å^2 was used for all image intervals.

The upper panel of Figure 3 illustrates the progression of the OIE algorithm in a six-dimensional case where only the two hydrogen atoms are free to move. Both the initial and final state are included in the same cut of the energy surface as illustrated in Figure 2, but the locations of the intermediate images and the training data points in Figure 3 need to be interpreted as projections due to small rotations and translations of the H_2 molecule on the plane parallel to the Cu(110) surface. The GP approximation based on covariance function $k_{1/r}$ looks surprisingly realistic already in the beginning, when the training data include only the energy and its first derivatives at the two end points and one intermediate image and the Hessian data at the end points. Before moving the images, however, the new early stopping rule requires one more image of the initial path to be evaluated in order for all the images to be within the allowed region. The NEB relaxations in the following three GPR iterations also end

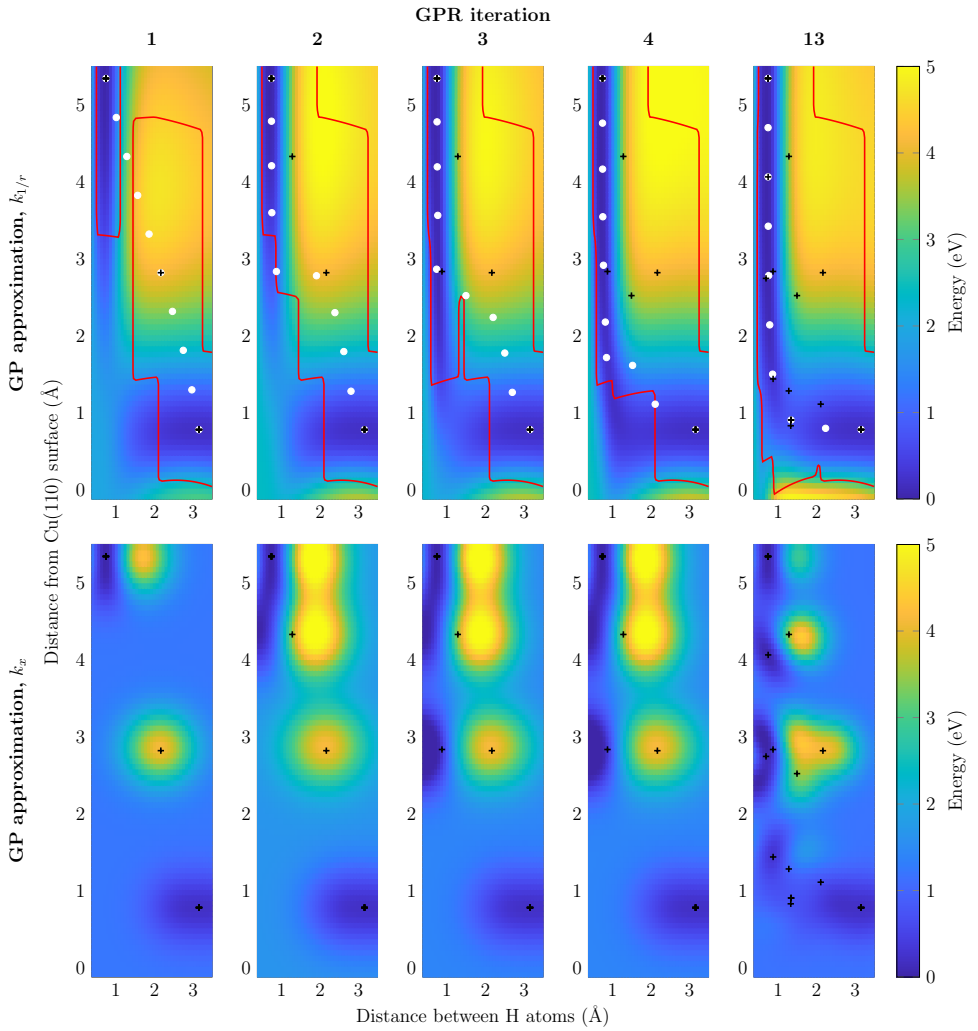

Figure 3: A two-dimensional cut through the potential energy surface for an H_2 molecule dissociating on a Cu(110) surface. The H–H molecular axis is parallel to the surface and perpendicular to the atom rows on the Cu(110) surface. The horizontal axis represents the distance between the two H atoms, and the vertical axis represents the distance between the H atoms and the Cu(110) surface. Upper panel: GP approximations with covariance function $k_{1/r}$ after one, two, three, four, and thirteen GPR iterations of the improved GP-NEB algorithm. The + signs mark projections of locations where energy and forces have been evaluated. The red line shows the border of the region allowed by the new early stopping rule, and the white dots are projections of the images at the end of each NEB relaxation phase. In the first four GPR iterations, the NEB relaxation phase is terminated by the early stopping rule. A converged MEP is obtained after thirteen GPR iterations. Lower panel: For comparison, GP approximations obtained with optimized hyperparameters for the stationary covariance function k_x are presented using the same training data sets as in the upper panel.

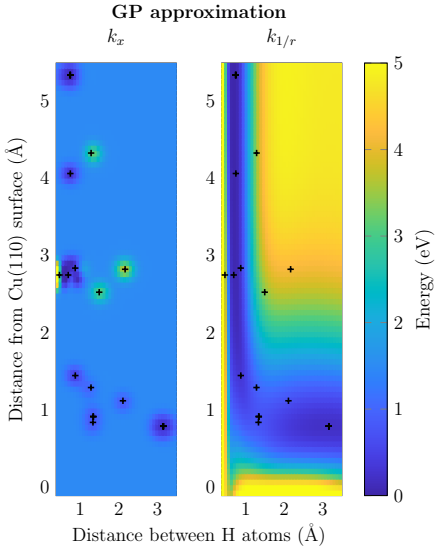

Figure 4: Illustrations of GP approximations based on covariance functions k_x (left) and $k_{1/r}$ (right), corresponding to the rightmost graphs in Figure 3 after adding one high-gradient training data point near the left border of the graph and reoptimizing the hyperparameters.

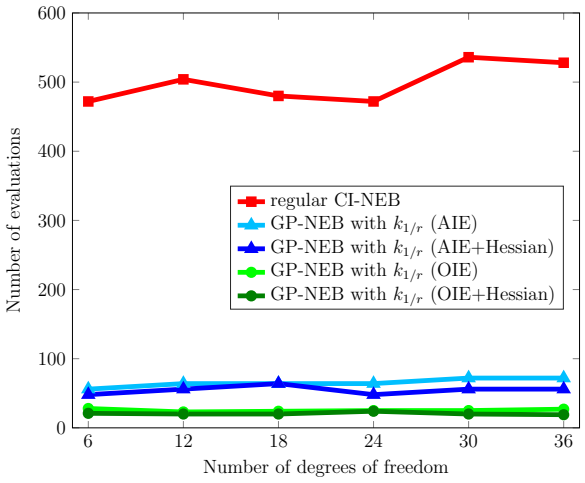

Figure 5: Number of energy and force evaluations required for convergence of CI-NEB calculations in the H_2/Cu(110) example as a function of the number of degrees of freedom, increased by allowing a larger number of Cu atoms to move. The performance of the all-images-evaluated (AIE) algorithm is presented by blue triangles and the performance of the one-image-evaluated (OIE) algorithm with green dots. The use of Hessian data at the initial and final state minima is indicated by darker color. All the GP-NEB results were obtained using the improved covariance function $k_{1/r}$ and the new stopping criterion.

up being terminated by the early stopping rule. Given how good the first prediction looks, the early stopping criterion may seem unnecessarily conservative, but it ensures that the relevant region is obtained safely with a reasonable number of energy and force evaluations, without the risk of getting too deep into regions of large atomic forces. The converged MEP is obtained after thirteen GPR iterations, and the convergence is then confirmed by one more evaluation per image.

For comparison, the lower panel of Figure 3 shows what the GP approximation with the same training data would look like if covariance function k_x based on the regular difference measure $\mathcal{D}_x$ is used instead of $k_{1/r}$. Note that the training data here consist of configurations where the energy surface with respect to the atom coordinates is still smooth enough to be interpolated with a reasonable stationary length scale. However, since the stationary covariance function extrapolates the attractive forces acting on the H atoms deep into the regions where the atoms collide or even pass through each other, it would be difficult to keep the images away from regions of large repulsive forces without too restrictive stopping rule. As shown in Figure 4, an additional data point from the repulsive region would make interpolation of the training data set more difficult and lead to a short length scale. For covariance function $k_{1/r}$, instead, this high-gradient data point would not cause problems.

Figure 5 shows the number of energy and force evaluations required for convergence of GP-NEB calculations where the six-dimensional configuration space was extended by allowing also the nearest Cu atoms to move. The corresponding results for the regular CI-NEB method were obtained using the velocity projection optimizer with a time step of 0.1 fs, which performed better than the L-BFGS optimizer in this example. The difference in the obtained saddle point energy between GP-NEB and regular CI-NEB was not larger than 0.0001 eV in any of the cases. Compared to the reference method, the number of evaluations is reduced by an order of magnitude when using the OIE algorithm with the improved covariance function $k_{1/r}$ and the new stopping criterion. The differences in the results between OIE and AIE algorithms and the effect of using the Hessian data at the initial and final state minima are quite similar to the earlier results for the heptamer benchmark obtained with the original formulation of the GP-NEB method.[7] For the reasons explained above, the original formulation based on the stationary squared exponential covariance function k_x could not be successfully applied to the H_2/Cu(110) system.

3.2 Application to H_2O diffusion on ice surface

Another example of an application challenging for the original formulation of the GP-NEB method, involving both strong intramolecular forces and weak intermolecular forces, is a diffusion hop of an H_2O admolecule on a (0001) surface of proton-disordered ice Ih. The slab representing the surface is here composed of 192 constrained water molecules arranged in four bilayers, and the energy surface is described by the TIP4P/2005f potential function,[34] which is a flexible version of TIP4P/2005.[35] This potential function has previously been used to simulate surface diffusion on various ice Ih surfaces using long-time-scale adaptive kinetic Monte Carlo simulations, and additional information on the modeling can be found in refs 36 and 37.

CI-NEB calculations for the transition were performed using a linear initial path, a spring constant of 10 eV/Å^2, and the same convergence thresholds as in the H_2/Cu(110) example ($T_{\mathrm{CI}} = 0.01$ eV/Å, $T_{\mathrm{MEP}} = 0.3$ eV/Å, $T^{\mathrm{GP}}_{\mathrm{CIon}} = 1$ eV/Å). In regular CI-NEB calculations, the L-BFGS optimizer performed better than the velocity projection optimizer for which a time step of 0.05 fs worked best. Figure 6 shows the minimum energy path obtained for the transition using the revised GP-NEB method. The energy difference between the initial state and saddle point was 0.054 eV with the regular CI-NEB method, 0.047 eV with the OIE version of the GP-NEB algorithm and 0.045 eV with the AIE version. While the regular CI-NEB calculation

required 1574 energy and force evaluations to reach convergence, the OIE version of the GP-NEB method converged with 35 and the AIE version with 90 evaluations. Thus, also for this molecular system, the GP-NEB method significantly reduces the number of evaluations.

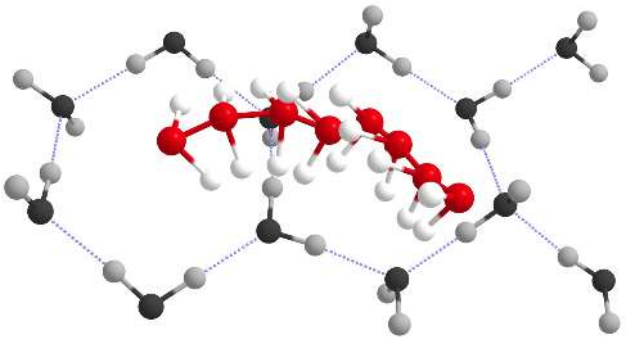

Figure 6: Minimum energy path for a diffusion hop of an H_2O admolecule on proton-disordered ice Ih(0001) surface calculated using the improved GP-NEB method. The O atom of the diffusing molecule is shown in red and the molecules in the surface bilayer in grayscale. Lower bilayer molecules are not presented. Hydrogen bonds are shown with dotted lines. The use of Gaussian process regression reduces the number of energy and force evaluations by more than an order of magnitude.

3.3 Application to the heptamer island benchmark

In an earlier publication,[7] the original formulation of the GP-NEB method based on the stationary covariance function k_x was shown to work well for a benchmark involving rearrangements of a heptamer island on a (111) surface of a face-centered cubic (FCC) crystal.[14,15] We now show that the improved covariance function $k_{1/r}$ based on the inverse-distance difference measure $\mathcal{D}_{1/r}$ gives even better performance in that the number of energy and force evaluations needed to reach convergence is reduced further. The initial, saddle point, and final state configurations for the thirteen transitions are shown in ref 7. In the initial state, the seven atoms sit at FCC surface sites and form a compact island. In two of the transitions, the whole island is shifted to hexagonal close-packed (HCP) sites on the surface. In some of the other transitions, a pair of edge atoms slides to adjacent FCC sites, an atom half way dissociates from the island, or one of the atoms is displaced away from the island while another one takes its place. The system is described by 343 platinum atoms with 56 atoms in each of the six layers, and the interactions between the atoms are described by a Morse potential.[14]

New GP-NEB calculations for the benchmark transitions were performed using the improved covariance function $k_{1/r}$ and the new early stopping criterion with the same settings as in the earlier tests:[7] An IDPP path with five intermediate images ($N_{\mathrm{im}} = 7$) was used as the initial path, the spring constant was set to 1 eV/Å^2 for all image intervals, and the convergence thresholds were the same that were used also for the H_2/Cu(110) and H_2O applications ($T_{\mathrm{CI}} = 0.01$ eV/Å, $T_{\mathrm{MEP}} = 0.3$ eV/Å, $T^{\mathrm{GP}}_{\mathrm{CIon}} = 1$ eV/Å). All platinum atoms were treated as the same atom type, and thus a common length scale was shared by all atom pairs in the system when calculating the inverse-distance difference measure $\mathcal{D}_{1/r}$ between configurations. The number of degrees of freedom was altered from 21 to 39 by allowing some of the nearest substrate atoms to move with the seven island atoms. In all cases, the saddle point energy differed less than 0.0004 eV from the regular CI-NEB result.

The average number of energy and force evaluations required in the new GP-NEB calculations as a function of the number of degrees of freedom is shown in Figure 7 with thick solid lines. The results are presented for both the OIE (green) and AIE (blue) algorithms with

(darker color) and without (lighter color) use of the Hessian data at the initial and final state minima. Depending on the algorithm variant, the improvements to the GP-NEB method reduce the number of required energy and force evaluations by about 30–50% compared to the earlier results (narrow dashed lines).

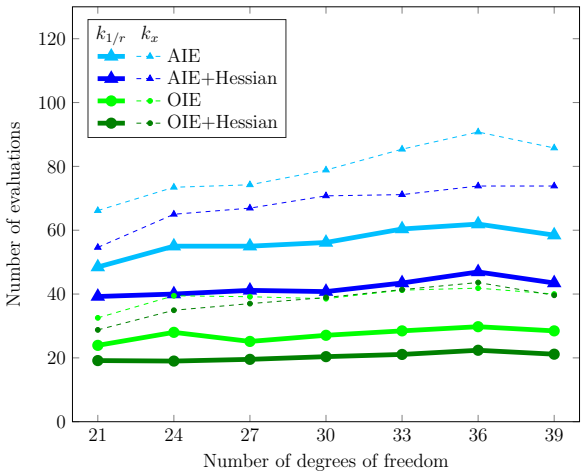

Figure 7: Number of energy and force evaluations required for convergence of CI-NEB calculations in the heptamer island benchmark with variants of the GP-NEB method. The average over the thirteen different transitions is presented as a function of the number of degrees of freedom, increased by allowing a larger number of substrate atoms to move. The narrow dashed lines present the earlier GP-NEB results[7] obtained using the stationary squared exponential covariance function k_x, and the thick solid lines present the corresponding results when using the improved covariance function $k_{1/r}$ and the new stopping criterion. The performance of the all-images-evaluated (AIE) algorithm is presented by blue triangles and the performance of the one-image-evaluated (OIE) algorithm by green dots. The use of Hessian data at the initial and final state minima is indicated by darker color.

4 Discussion

The examples of application of the GP-NEB method studied here show that interpolation of the energy surface with respect to atom coordinates may be difficult with a stationary Gaussian process covariance function that has the same characteristic length scale throughout the coordinate space. An improved covariance function was presented here, where the similarity between two configurations is based on differences in inverted interatomic distances within each of the two configurations. The closer two atoms are to each other, the larger effect a small displacement of these atoms toward or away from each other has on the inverse-distance difference measure. This makes the covariance function nonstationary with respect to the atom coordinates and the energy surface easier to represent by the Gaussian process model.

The justification of the inverse-distance covariance function is based on the assumption that the energy of the system can be presented as a smooth function of interatomic distances. In other words, if there are two configurations with the same interatomic distances, also the energy should be the same. Since the covariance function gives almost full correlation for the two energy values, reduced only by the small noise variance σ^2, problems may arise if the energies differ by significantly more than σ. Therefore, if a cutoff distance is used to reduce the number of atom pairs taken into account in the covariance function, the changes in energy outside the cutoff distance should be kept comparable to σ. Similar problems may emerge if the energy evaluations

involve periodic boundary conditions not taken into account when calculating the interatomic distances for the covariance function. In a proper treatment of such systems, the contribution of an interatomic distance should be suppressed smoothly to zero before half cell size is reached in any direction in order to avoid discontinuities in the derivatives of the difference measure with respect to the original coordinates.

As mentioned in the Methods section and illustrated in the SI, a stationary model becomes to some extent more flexible if the smoothness assumptions are loosened by replacing the infinitely differentiable squared exponential covariance function with an appropriate member of the Matérn family. The improved covariance function based on the inverse-distance difference measure could similarly be made more flexible if the difference measure was fed to a Matérn covariance function. In the examples presented in this article, however, this covariance function worked best in the squared exponential form. This indicates that the energy was behaving smoothly enough with respect to the inverted interatomic distances, in order to be successfully modeled with an infinitely differentiable covariance function.

In addition to the inverse interatomic distances, it is possible to include also angles between the lines connecting the atoms when defining the similarity between configurations. This would, however, require handling triplets of atoms, which would complicate and slow down calculation of the covariances. In principle, the GP-NEB method can also be combined with more complicated approximative models of local atomic environments as those used in the GAP potentials.[12,13] Our goal, however, has been to keep the model simple with respect to the atom coordinates and general enough to be able to interpolate the surroundings of minimum energy paths accurately without extensive tuning.

Besides modifying the covariance function, an early stopping criterion restricting relative changes in the interatomic distances during the NEB relaxation was introduced. The purpose of the new stopping rule is to avoid unphysical configurations that may disturb the fitting of the GP model and also to generally stabilize the development of the model by constraining how far the NEB images can move into unexplored regions. However, since the criterion is only based on interatomic distances, it does not necessarily restrict joint movement of a group of atoms. To restrict also joint movement of atoms, we considered one more early stopping criterion based on the displacement of each atom scaled by the distance to the nearest atom. This criterion would similarly require that there exists an evaluated data point that fulfils the condition for all atoms. However, we did not find this addition useful in the examples presented here. Rather than stabilizing the algorithm, it increased the number of evaluations by triggering unnecessary energy and force evaluations.

The advantage of the GP-NEB method relies on the assumption that training of the GP model and evaluations on the approximated energy surface can be performed in negligible time compared to accurate energy and force evaluations. In practice, however, the cost of the GP approximation limits the applicability of the method to systems with around a few dozen moving atoms or less. The computational bottleneck of a standard implementation of Gaussian process regression is the inversion of the training covariance matrix with a cubic time requirement and a quadratic memory requirement with respect to the length of the observation vector. A recently introduced approach[38,39] avoids explicit inversion of the covariance matrix and thereby reduces the scaling of the training time from cubic to quadratic and the scaling of the memory requirement from quadratic to linear without compromising the accuracy of the inference. Since the approach is also parallelizable, further acceleration is possible by using multiple processors. When the training data set includes derivatives with respect to all $3N_\mathrm{m}$ input coordinates, this approach would mean quadratic scaling with respect to both the number of data points, N, and the number of moving atoms, N_m. The construction of the matrix requires evaluations of $(N(1+3N_\mathrm{m}))^2$ covariances and the prediction of the whole gradient vector N_m derivatives of $N(1+3N_\mathrm{m})$ covariances. Even though calculation of any of these elements requires evaluation

of the difference measure, which here includes a sum over all pairs of moving atoms, this needs to be done only once for each pair of data points. By storing the value of the difference measure and its derivative with respect to each input coordinate while building each of the N^2 blocks, the whole covariance matrix can be built in $\mathcal{O}(N^2 N_\mathrm{m}^2)$ time, and similarly, the prediction of the whole gradient vector can be done in $\mathcal{O}(N N_\mathrm{m}^2)$ time. Thus, even though the inverse-distance formulation increases the cost of individual covariance function evaluations, it does not affect the scaling of the cost of the whole algorithm.

If found necessary, practical speedup could be obtained by reducing the training data set by selectively ignoring some of the data points, derivatives of some data points or derivatives with respect to movement of some atoms. It would also be possible to train a separate GP model for each image using different training data sets. If the evaluations of the GP approximation are taking much time, it might be convenient to reduce the maximum number of inner iterations and start the following NEB relaxation phase where the previous one ended. The optimization of the hyperparameters could be as well started from the previous values after a few initial rounds or even skipped for some number of rounds after the values have stabilized, and it is also possible to use the same length scale for all atom pair types. One possible approach would be to start with a lighter approximate model with larger noise assumed and switch to a noiseless model when converging to the minimum energy path.

Supporting Information

- Extensions of Figures 1–4 and 7 including GP approximations obtained with stationary Matérn covariance functions and GP-NEB results obtained by feeding the inverse-distance difference measure to Matérn covariance functions (PDF)

Funding

This work was financially supported by the Academy of Finland (Grant 278260; Flagship programme: Finnish Center for Artificial Intelligence, FCAI, Grants 320181, 320182, 320183). O.-P.K. was supported by the Finnish Cultural Foundation (Grant 180536) and V.Á. by a doctoral fellowship from the University of Iceland Research Fund.

Acknowledgements

Computational resources were provided by the Aalto Science-IT project and the Icelandic High Performance Computing services located at the University of Iceland.

References

[1] Mills, G.; Jónsson, H.; Schenter, G. K. Reversible work based transition state theory: application to H_2 dissociative adsorption. *Surf. Sci.* **1995**, 324, 305.

[2] Jónsson, H.; Mills, G.; Jacobsen, K. W. Nudged elastic band method for finding minimum energy paths of transitions. In *Classical and Quantum Dynamics in Condensed Phase Simulations*; Berne, B. J., Ciccotti, G., Coker, D. F., Eds.; World Scientific: Singapore, 1998; pp 385–404.

[3] Peterson, A. A. Acceleration of saddle-point searches with machine learning. *J. Chem. Phys.* **2016**, 145, 74106.

[4] Behler, J. Atom-centered symmetry functions for constructing high-dimensional neural network potentials. *J. Chem. Phys.* **2011**, 134, 74106.

[5] Artrith, N.; Morawietz, T.; Behler, J. High-dimensional neural-network potentials for multicomponent systems: applications to zinc oxide. *Phys. Rev. B* **2011**, 83, 153101.

[6] Koistinen, O.-P.; Maras, E.; Vehtari, A.; Jónsson, H. Minimum energy path calculations with Gaussian process regression. *Nanosyst.: Phys. Chem. Math.* **2016**, 7, 925. A slightly corrected version available as e-print arXiv:1703.10423.

[7] Koistinen, O.-P.; Dagbjartsdóttir, F. B.; Ásgeirsson, V.; Vehtari, A.; Jónsson, H. Nudged elastic band calculations accelerated with Gaussian process regression. *J. Chem. Phys.* **2017**, 147, 152720.

[8] O'Hagan, A. Curve fitting and optimal design for prediction. *J. Royal Stat. Soc. B* **1978**, 40, 1.

[9] MacKay, D. J. C. Introduction to Gaussian processes. In *Neural Networks and Machine Learning*; Bishop, C. M., Ed.; Springer-Verlag: Berlin, 1998; pp 133–166.

[10] Neal, R. M. Regression and classification using Gaussian process priors. In *Bayesian Statistics 6*; Bernardo, J. M., Berger, J. O., Dawid, A. P., Smith, A. F. M., Eds.; Clarendon Press: Oxford, 1999; pp 475–501.

[11] Rasmussen, C. E.; Williams, C. K. I. *Gaussian Processes for Machine Learning*; MIT Press: Cambridge, 2006.

[12] Bartók, A. P.; Payne, M. C.; Condor, R.; Csányi, G. Gaussian approximation potentials: the accuracy of quantum mechanics, without the electrons. *Phys. Rev. Lett.* **2010**, 104, 136403.

[13] Bartók, A. P.; Csányi, G. Gaussian approximation potentials: a brief tutorial introduction. *Int. J. Quantum Chem.* **2015**, 115, 1051.

[14] Henkelman, G.; Jóhannesson, G. H.; Jónsson, H. Methods for finding saddle points and minimum energy paths. In *Theoretical Methods in Condensed Phase Chemistry*; Schwartz, S. D., Ed.; Progress in Theoretical Chemistry and Physics 5; Kluwer Academic: New York, 2000; pp 269–300.

[15] Chill, S. T.; Stevenson, J.; Ruhle, V.; Shang, C.; Xiao, P.; Farrell, J. D.; Wales, D. J.; Henkelman, G. Benchmarks for characterization of minima, transition states and pathways in atomic, molecular, and condensed matter systems. *J. Chem. Theory Comput.* **2014**, 10, 5476.

[16] Garrido Torres, J. A.; Jennings, P. C.; Hansen, M. H.; Boes, J. R.; Bligaard, T. Low-scaling algorithm for nudged elastic band calculations using a surrogate machine learning model. *Phys. Rev. Lett.* **2019**, 122, 156001.

[17] Smidstrup, S.; Pedersen, A.; Stokbro, K.; Jónsson, H. Improved initial guess for minimum energy path calculations. *J. Chem. Phys.* **2014**, 140, 214106.

[18] Zhu, X.; Thompson, K. C.; Martínez, T. J. Geodesic interpolation for reaction pathways. *J. Chem. Phys.* **2019**, 150, 164103.

[19] Henkelman, G.; Jónsson, H. Improved tangent estimate in the nudged elastic band method for finding minimum energy paths and saddle points. *J. Chem. Phys.* **2000**, 113, 9978.

[20] Henkelman, G.; Uberuaga, B. P.; Jónsson, H. A climbing image nudged elastic band method for finding saddle points and minimum energy paths. *J. Chem. Phys.* **2000**, 113, 9901.

[21] Swope, W. C.; Andersen, H. C.; Berens, P. H.; Wilson, K. R. A computer simulation method for the calculation of equilibrium constants for the formation of physical clusters of molecules: application to small water clusters. *J. Chem. Phys.* **1982**, 76, 637.

[22] Nocedal, J. Updating quasi-Newton matrices with limited storage. *Math. Comput.* **1980**, 35, 773.

[23] Sheppard, D.; Terrell, R.; Henkelman, G. Optimization methods for finding minimum energy paths. *J. Chem. Phys.* **2008**, 128, 134106.

[24] Chill, S. T.; Welborn, M.; Terrell, R.; Zhang, L.; Berthet, J.-C.; Pedersen, A.; Jónsson, H.; Henkelman, G. EON: software for long time simulations of atomic scale systems. *Model. Simul. Mater. Sci. Eng.* **2014**, 22, 55002.

[25] Matérn, B. *Spatial variation*; Allmänna förlaget: Stockholm, 1960.

[26] Denzel, A.; Kästner, J. Gaussian process regression for geometry optimization. *J. Chem. Phys.* **2018**, 148, 94114.

[27] O'Hagan, A. Some Bayesian numerical analysis. In *Bayesian Statistics 4*; Bernardo, J. M., Berger, J. O., Dawid, A. P., Smith, A. F. M., Eds.; Clarendon Press: Oxford, 1992; pp 345–363.

[28] Rasmussen, C. E. Gaussian processes to speed up hybrid Monte Carlo for expensive Bayesian integrals. In *Bayesian Statistics 7*; Bernardo, J. M., Dawid, A. P., Berger, J. O., West, M., Heckerman, D., Bayarri, M. J., Smith, A. F. M., Eds.; Clarendon Press: Oxford, 2003; pp 651–659.

[29] Solak, E.; Murray-Smith, R.; Leithead, W. E.; Leith, D. J.; Rasmussen, C. E. Derivative observations in Gaussian process models of dynamic systems. In *Advances in Neural Information Processing Systems 15*; Becker, S., Thrun, S., Obermayer, K., Eds.; MIT Press: Cambridge, 2003; pp 1057–1064.

[30] Riihimäki, J.; Vehtari, A. Gaussian processes with monotonicity information. In *Proceedings of the Thirteenth International Converence on Artificial Intelligence and Statistics*; Teh, Y. W., Titterington, M., Eds.; Proceedings of Machine Learning Research, 2010; pp 645–652.

[31] Vanhatalo, J.; Riihimäki, J.; Hartikainen, J.; Jylänki, P.; Tolvanen, V.; Vehtari, A. GPstuff: Bayesian modeling with Gaussian processes. *J. Mach. Learn. Res.* **2013**, 14, 1175.

[32] Bishop, C. M. *Neural Networks for Pattern Recognition*; Clarendon Press: Oxford, 1995; pp 282–285.

[33] Daw, M. S.; Baskes, M. I. Embedded-atom method: derivation and application to impurities, surfaces, and other defects in metals. *Phys. Rev. B* **1984**, 29, 6443.

[34] González, M. A.; Abascal, J. L. F. A flexible model for water based on TIP4P/2005. *J. Chem. Phys.* **2011**, 135, 224516.

[35] Abascal, J. L. F.; Vega, C. A general purpose model for the condensed phases of water: TIP4P/2005. *J. Chem. Phys.* **2005**, 123, 234505.

[36] Pedersen, A.; Wikfeldt, K. T.; Karssemeijer, L.; Cuppen, H.; Jónsson, H. Molecular reordering processes on ice (0001) surfaces from long timescale simulations. *J. Chem. Phys.* **2014**, 141, 234706.

[37] Pedersen, A.; Karssemeijer, L.; Cuppen, H. M.; Jónsson, H. Long-time-scale simulations of H_2O admolecule diffusion on ice Ih(0001) surfaces. *J. Phys. Chem. C* **2015**, 119, 16528.

[38] Gardner, J. R.; Pleiss, G.; Weinberger, K. Q.; Bindel, D.; Wilson, A. G. GPyTorch: blackbox matrix-matrix Gaussian process inference with GPU acceleration. In *Advances in Neural Information Processing Systems 31*; Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R., Eds.; Curran Associates: Red Hook, 2018; pp 7576–7586.

[39] Wang, K. A.; Pleiss, G.; Gardner, J. R.; Tyree, S.; Wilson, A. G. Exact Gaussian processes on a million data points. Preprint arXiv:1903.08114, 2019.

Appendix

Partial derivatives of covariance function k_x

When predicting derivatives of a function modeled with a Gaussian process (eq 9) or when dealing with derivative data in the training data set (eqs 10–11), partial derivatives of the covariance function with respect to input coordinates are required. To calculate the partial derivatives of covariance function k_x, defined in eq 2, we first calculate the partial derivative of the square of the regular difference measure $\mathcal{D}_x(\mathbf{x}, \mathbf{x}')$, defined in eq 3, with respect to the d^{th} coordinate of moving atom i in $\mathbf{x}$,

$$\frac{\partial \mathcal{D}_x^2(\mathbf{x}, \mathbf{x}')}{\partial x_{i,d}} = \frac{2(x_{i,d} - x'_{i,d})}{l^2}, \tag{15}$$

and with respect to both the d_1^{th} coordinate of moving atom i_1 in $\mathbf{x}$ and d_2^{th} coordinate of moving atom i_2 in $\mathbf{x}'$,

$$\frac{\partial^2 \mathcal{D}_x^2(\mathbf{x}, \mathbf{x}')}{\partial x_{i_1,d_1} \partial x'_{i_2,d_2}} = \begin{cases} 0, & \text{if } i_1 \neq i_2 \vee d_1 \neq d_2 \\ \dfrac{-2}{l^2}, & \text{if } i_1 = i_2 \wedge d_1 = d_2. \end{cases} \tag{16}$$

Using chain rules, the corresponding partial derivatives of covariance function k_x can be presented as

$$\frac{\partial k_x(\mathbf{x}, \mathbf{x}')}{\partial x_{i,d}} = \frac{\partial k_x(\mathbf{x}, \mathbf{x}')}{\partial \mathcal{D}_x^2(\mathbf{x}, \mathbf{x}')} \cdot \frac{\partial \mathcal{D}_x^2(\mathbf{x}, \mathbf{x}')}{\partial x_{i,d}} \tag{17}$$

and

$$\frac{\partial^2 k_x(\mathbf{x}, \mathbf{x}')}{\partial x_{i_1,d_1} \partial x'_{i_2,d_2}} = \frac{\partial k_x(\mathbf{x}, \mathbf{x}')}{\partial \mathcal{D}_x^2(\mathbf{x}, \mathbf{x}')} \cdot \frac{\partial^2 \mathcal{D}_x^2(\mathbf{x}, \mathbf{x}')}{\partial x_{i_1,d_1} \partial x'_{i_2,d_2}} + \frac{\partial^2 k_x(\mathbf{x}, \mathbf{x}')}{\partial (\mathcal{D}_x^2(\mathbf{x}, \mathbf{x}'))^2} \cdot \frac{\partial \mathcal{D}_x^2(\mathbf{x}, \mathbf{x}')}{\partial x_{i_1,d_1}} \cdot \frac{\partial \mathcal{D}_x^2(\mathbf{x}, \mathbf{x}')}{\partial x'_{i_2,d_2}}, \tag{18}$$

where

$$\frac{\partial k_x(\mathbf{x}, \mathbf{x}')}{\partial \mathcal{D}_x^2(\mathbf{x}, \mathbf{x}')} = -\frac{\sigma_{\text{m}}^2}{2} \exp\left(-\frac{1}{2}\mathcal{D}_x^2(\mathbf{x}, \mathbf{x}')\right) \tag{19}$$

and

$$\frac{\partial^2 k_x(\mathbf{x}, \mathbf{x}')}{\partial (\mathcal{D}_x^2(\mathbf{x}, \mathbf{x}'))^2} = \frac{\sigma_{\text{m}}^2}{4} \exp\left(-\frac{1}{2}\mathcal{D}_x^2(\mathbf{x}, \mathbf{x}')\right) \tag{20}$$

are the first and second derivatives of the covariance function with respect to the squared difference measure.

When optimizing the hyperparameters, it is useful to differentiate the covariance function and its derivatives also with respect to the hyperparameters. Differentiation with respect to magnitude σ_{m} is trivial, since σ_{m}^2 can be factorized out from the expressions. With respect to the isotropic length scale l, we start again by differentiating the squared difference measure and its derivatives:

$$\frac{\partial \mathcal{D}_x^2(\mathbf{x}, \mathbf{x}')}{\partial l} = \sum_{i=1}^{N_{\text{m}}} \sum_{d=1}^{3} \frac{-2(x_{i,d} - x'_{i,d})^2}{l^3}, \tag{21}$$

$$\frac{\partial \mathcal{D}_x^2(\mathbf{x}, \mathbf{x}')}{\partial x_{i,d} \partial l} = \frac{-4(x_{i,d} - x'_{i,d})}{l^3}, \tag{22}$$

$$\frac{\partial^2 \mathcal{D}_x^2(\mathbf{x}, \mathbf{x}')}{\partial x_{i_1,d_1} \partial x'_{i_2,d_2} \partial l} = \begin{cases} 0, & \text{if } i_1 \neq i_2 \vee d_1 \neq d_2 \\ \dfrac{4}{l^3}, & \text{if } i_1 = i_2 \wedge d_1 = d_2. \end{cases} \tag{23}$$

Using chain rules, we can now differentiate the covariance function and its derivatives:

$$\frac{\partial k_x(\mathbf{x},\mathbf{x}')}{\partial l_\psi} = \frac{\partial k_x(\mathbf{x},\mathbf{x}')}{\partial \mathcal{D}_x^2(\mathbf{x},\mathbf{x}')} \cdot \frac{\partial \mathcal{D}_x^2(\mathbf{x},\mathbf{x}')}{\partial l}, \tag{24}$$

$$\frac{\partial^2 k_x(\mathbf{x},\mathbf{x}')}{\partial x_{i,d}\partial l} = \frac{\partial k_x(\mathbf{x},\mathbf{x}')}{\partial \mathcal{D}_x^2(\mathbf{x},\mathbf{x}')} \cdot \frac{\partial^2 \mathcal{D}_x^2(\mathbf{x},\mathbf{x}')}{\partial x_{i,d}\partial l} + \frac{\partial^2 k_x(\mathbf{x},\mathbf{x}')}{\partial (\mathcal{D}_x^2(\mathbf{x},\mathbf{x}'))^2} \cdot \frac{\partial \mathcal{D}_x^2(\mathbf{x},\mathbf{x}')}{\partial x_{i,d}} \cdot \frac{\partial \mathcal{D}_x^2(\mathbf{x},\mathbf{x}')}{\partial l}, \tag{25}$$

$$\begin{aligned}\frac{\partial^3 k_x(\mathbf{x},\mathbf{x}')}{\partial x_{i_1,d_1}\partial x'_{i_2,d_2}\partial l} &= \frac{\partial k_x(\mathbf{x},\mathbf{x}')}{\partial \mathcal{D}_x^2(\mathbf{x},\mathbf{x}')} \cdot \frac{\partial^3 \mathcal{D}_x^2(\mathbf{x},\mathbf{x}')}{\partial x_{i_1,d_1}\partial x'_{i_2,d_2}\partial l} \\ &+ \frac{\partial^2 k_x(\mathbf{x},\mathbf{x}')}{\partial (\mathcal{D}_x^2(\mathbf{x},\mathbf{x}'))^2} \cdot \left(\frac{\partial \mathcal{D}_x^2(\mathbf{x},\mathbf{x}')}{\partial x_{i_1,d_1}} \cdot \frac{\partial^2 \mathcal{D}_x^2(\mathbf{x},\mathbf{x}')}{\partial x'_{i_2,d_2}\partial l} \right. \\ &\left. + \frac{\partial \mathcal{D}_x^2(\mathbf{x},\mathbf{x}')}{\partial x'_{i_2,d_2}} \cdot \frac{\partial^2 \mathcal{D}_x^2(\mathbf{x},\mathbf{x}')}{\partial x_{i_1,d_1}\partial l} + \frac{\partial \mathcal{D}_x^2(\mathbf{x},\mathbf{x}')}{\partial l} \cdot \frac{\partial^2 \mathcal{D}_x^2(\mathbf{x},\mathbf{x}')}{\partial x_{i_1,d_1}\partial x'_{i_2,d_2}} \right) \\ &+ \frac{\partial^3 k_x(\mathbf{x},\mathbf{x}')}{\partial (\mathcal{D}_x^2(\mathbf{x},\mathbf{x}'))^3} \cdot \frac{\partial \mathcal{D}_x^2(\mathbf{x},\mathbf{x}')}{\partial x_{i_1,d_1}} \cdot \frac{\partial \mathcal{D}_x^2(\mathbf{x},\mathbf{x}')}{\partial x'_{i_2,d_2}} \cdot \frac{\partial \mathcal{D}_x^2(\mathbf{x},\mathbf{x}')}{\partial l},\end{aligned} \tag{26}$$

where

$$\frac{\partial^3 k_x(\mathbf{x},\mathbf{x}')}{\partial (\mathcal{D}_x^2(\mathbf{x},\mathbf{x}'))^3} = -\frac{\sigma_{\mathrm{m}}^2}{8} \exp\left(-\frac{1}{2}\mathcal{D}_x^2(\mathbf{x},\mathbf{x}')\right). \tag{27}$$

Partial derivatives of covariance function $k_{1/r}$

The partial derivative of the square of the inverse-distance difference measure $\mathcal{D}_{1/r}(\mathbf{x},\mathbf{x}')$, defined in eq 4, with respect to the d^{th} coordinate of moving atom i in $\mathbf{x}$ is given by

$$\frac{\partial \mathcal{D}_{1/r}^2(\mathbf{x},\mathbf{x}')}{\partial x_{i,d}} = \sum_{\substack{j \in A_{\mathrm{m}} \setminus \{i\} \\ \vee \\ j \in A_{\mathrm{f}}}} \left[\frac{-2(x_{i,d} - x_{j,d})}{l_{\phi(i,j)}^2 r_{i,j}^3(\mathbf{x})} \left(\frac{1}{r_{i,j}(\mathbf{x})} - \frac{1}{r_{i,j}(\mathbf{x}')} \right) \right], \tag{28}$$

and with respect to both the d_1^{th} coordinate of moving atom i_1 in $\mathbf{x}$ and d_2^{th} coordinate of moving atom i_2 in $\mathbf{x}'$ by

$$\frac{\partial^2 \mathcal{D}_{1/r}^2(\mathbf{x},\mathbf{x}')}{\partial x_{i_1,d_1}\partial x'_{i_2,d_2}} = \begin{cases} \dfrac{2(x_{i_1,d_1} - x_{i_2,d_1})(x'_{i_1,d_2} - x'_{i_2,d_2})}{l_{\phi(i_1,i_2)}^2 r_{i_1,i_2}^3(\mathbf{x}) r_{i_1,i_2}^3(\mathbf{x}')}, & \text{if } i_1 \neq i_2 \\ \displaystyle\sum_{\substack{j \in A_{\mathrm{m}} \setminus \{i\} \\ \vee \\ j \in A_{\mathrm{f}}}} \dfrac{-2(x_{i,d_1} - x_{j,d_1})(x'_{i,d_2} - x'_{j,d_2})}{l_{\phi(i,j)}^2 r_{i,j}^3(\mathbf{x}) r_{i,j}^3(\mathbf{x}')}, & \text{if } i_1 = i_2 = i. \end{cases} \tag{29}$$

The corresponding partial derivatives of covariance function $k_{1/r}$ can be presented with similar expressions as shown for k_x in eqs 17 and 18, keeping in mind that k_x and $k_{1/r}$ have the same derivatives with respect to the square of the difference measure.

Similarly, $k_{1/r}$ and its derivatives can be differentiated with respect to length scale l_ψ for atom pair type ψ using similar chain rules as shown in eqs 24, 25, and 26. The corresponding partial derivatives of the square of the difference measure $\mathcal{D}_{1/r}$, required for these expressions, are given by

$$\frac{\partial \mathcal{D}_{1/r}^2(\mathbf{x},\mathbf{x}')}{\partial l_\psi} = \sum_{i \in A_{\mathrm{m}}} \sum_{\substack{\left[\substack{j \in A_{\mathrm{m}}, j > i \\ \vee \\ j \in A_{\mathrm{f}}} \right] \\ \phi(i,j) = \psi}} \frac{-2\left(\frac{1}{r_{i,j}(\mathbf{x})} - \frac{1}{r_{i,j}(\mathbf{x}')}\right)^2}{l_\psi^3}, \tag{30}$$

$$\frac{\partial^2 \mathcal{D}^2_{1/r}(\mathbf{x},\mathbf{x}')}{\partial x_{i,d}\partial l_\psi} = \sum_{\substack{\left[\begin{smallmatrix} j\in A_{\mathrm{m}}\setminus\{i\} \\ \vee \\ j\in A_{\mathrm{f}} \end{smallmatrix}\right] \\ \phi(i,j)=\psi}} \left[\frac{4(x_{i,d}-x_{j,d})}{l_\psi^3 r_{i,j}^3(\mathbf{x})}\left(\frac{1}{r_{i,j}(\mathbf{x})}-\frac{1}{r_{i,j}(\mathbf{x}')}\right)\right], \tag{31}$$

and

$$\frac{\partial^3 \mathcal{D}^2_{1/r}(\mathbf{x},\mathbf{x}')}{\partial x_{i_1,d_1}\partial x'_{i_2,d_2}\partial l_\psi} = \begin{cases} 0, & \text{if } i_1 \neq i_2 \wedge \phi(i_1,i_2)\neq\psi \\ \dfrac{-4(x_{i_1,d_1}-x_{i_2,d_1})(x'_{i_1,d_2}-x'_{i_2,d_2})}{l_\psi^3 r_{i_1,i_2}^3(\mathbf{x}) r_{i_1,i_2}^3(\mathbf{x}')}, & \text{if } i_1 \neq i_2 \wedge \phi(i_1,i_2)=\psi \\ \displaystyle\sum_{\substack{\left[\begin{smallmatrix} j\in A_{\mathrm{m}}\setminus\{i\} \\ \vee \\ j\in A_{\mathrm{f}} \end{smallmatrix}\right] \\ \phi(i,j)=\psi}} \dfrac{4(x_{i,d_1}-x_{j,d_1})(x'_{i,d_2}-x'_{j,d_2})}{l_\psi^3 r_{i,j}^3(\mathbf{x}) r_{i,j}^3(\mathbf{x}')}, & \text{if } i_1 = i_2 = i. \end{cases} \tag{32}$$