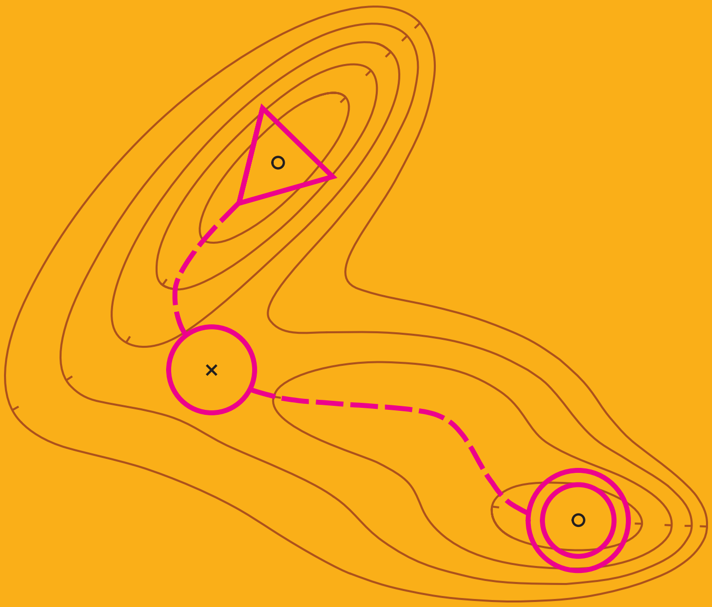
Department of Computer Science

Algorithms for Finding Saddle Points and Minimum Energy Paths Using Gaussian Process Regression

Olli-Pekka Koistinen



Aalto University

DOCTORAL DISSERTATIONS

Aalto University publication series
DOCTORAL DISSERTATIONS 222/2019

Algorithms for Finding Saddle Points and Minimum Energy Paths Using Gaussian Process Regression

Olli-Pekka Koistinen

A doctoral dissertation completed for the degree of Doctor of Science (Technology) to be defended, with the permission of the Aalto University School of Science, at a public examination held at the lecture hall E (Y124) of the school on 9 January 2020 at 12.

This doctoral dissertation has been conducted under a convention for the joint supervision at Aalto University (Finland) and University of Iceland (Iceland).

Aalto University
School of Science
Department of Computer Science
University of Iceland

Supervising professors

Professor Aki Vehtari, Aalto University, Finland
Professor Hannes Jónsson, University of Iceland, Iceland

Preliminary examiners

Professor Johannes Kästner, University of Stuttgart, Germany
Professor Andrew Peterson, Brown University, United States

Opponent

Professor Thomas Bligaard, Technical University of Denmark, Denmark

Aalto University publication series
DOCTORAL DISSERTATIONS 222/2019

ISBN 978-952-60-8850-1 (printed)
ISBN 978-952-60-8851-8 (pdf)
ISSN 1799-4934 (printed)
ISSN 1799-4942 (pdf)
http://urn.fi/URN:ISBN:978-952-60-8851-8

Unigrafia Oy
Helsinki 2019

Finland

NORDIC SWAN ECOLABEL
Printed matter
4041-0619

Aalto University

Abstract

Aalto University, P.O. Box 11000, FI-00076 Aalto www.aalto.fi

Author
Olli-Pekka Koistinen

Name of the doctoral dissertation
Algorithms for Finding Saddle Points and Minimum Energy Paths Using Gaussian Process Regression

Publisher School of Science

Unit Department of Computer Science

Series Aalto University publication series DOCTORAL DISSERTATIONS 222/2019

Field of research Computational science

Manuscript submitted 23 October 2019 **Date of the defence** 9 January 2020

Permission for public defence granted (date) 26 November 2019 **Language** English

□ **Monograph** ☒ **Article dissertation** □ **Essay dissertation**

Abstract

Chemical reactions and other transitions involving rearrangements of atoms can be studied theoretically by analyzing a potential energy surface defined in a high-dimensional space of atom coordinates. Local minimum points of the energy surface correspond to stable states of the system, and minimum energy paths connecting these states characterize mechanisms of possible transitions. Of particular interest is often the maximum point of the minimum energy path, which is located at a first-order saddle point of the energy surface and can be used to estimate the activation energy and rate of the particular transition.

Minimum energy paths and saddle points between two known states have been traditionally searched with iterative methods where a chain of discrete points of the coordinate space is moved and stretched towards a minimum energy path according to imaginary forces based on gradient vectors of the potential energy surface. The actual saddle point can be found by reversing the component of the gradient vector parallel to the path at one of the points of the chain and letting this point climb along the path towards the saddle point. If the end state of the transition is unknown, the saddle point can be searched correspondingly by rotating a pair of closely spaced points towards the orientation of the lowest curvature, reversing the gradient component corresponding to this direction, and moving the pair towards the saddle point. These methods may, however, require hundreds of iterations, and since accurate evaluation of the gradient vector is often computationally expensive, the information obtained from previous iterations should be utilized as efficiently as possible to decrease the number of iterations. Using statistical models, an approximation to the energy surface can be constructed, and a minimum energy path or a saddle point can be searched on the approximate surface. The accuracy of the solution can be checked with further evaluations, which can be then used to update the model for following iterations.

In this dissertation, machine learning algorithms based on Gaussian process regression are developed to enhance searches of minimum energy paths and saddle points. Gaussian process models serve here as flexible prior probability models for potential energy surfaces. Observed values of both energy and its derivatives can be used to update the model, and the posterior predictive distribution obtained as a result of Bayesian inference provides also an uncertainty estimate, which can be utilized when selecting new observation points. Separate methods are presented both for finding a minimum energy path between two known states and a saddle point located in the vicinity of a given start point. Based on simple test examples, the methods utilizing Gaussian processes may reduce the number of evaluations to a fraction of what is required by conventional methods.

Keywords saddle point, minimum energy path, Gaussian process, machine learning

ISBN (printed) 978-952-60-8850-1 **ISBN (pdf)** 978-952-60-8851-8

ISSN (printed) 1799-4934 **ISSN (pdf)** 1799-4942

Location of publisher Helsinki **Location of printing** Helsinki **Year** 2019

Pages 144 **urn** http://urn.fi/URN:ISBN:978-952-60-8851-8

Aalto-yliopisto

Aalto-yliopisto, PL 11000, 00076 Aalto www.aalto.fi

Tiivistelmä

Tekijä
Olli-Pekka Koistinen

Väitöskirjan nimi
Gaussisia prosesseja hyödyntäviä menetelmiä satulapisteiden ja minimienergiapolkujen etsintään

Julkaisija Perustieteiden korkeakoulu

Yksikkö Tietotekniikan laitos

Sarja Aalto University publication series DOCTORAL DISSERTATIONS 222/2019

Tutkimusala Laskennallinen tiede

Käsikirjoituksen pvm 23.10.2019 **Väitöspäivä** 09.01.2020

Väittelyluvan myöntämispäivä 26.11.2019 **Kieli** Englanti

□ **Monografia** ☒ **Artikkeliväitöskirja** □ **Esseeväitöskirja**

Tiivistelmä

Kemiallisia reaktioita ja muita atomien liikkeisiin perustuvia tapahtumia voidaan tarkastella teoreettisesti atomien koordinaattien muodostamassa moniulotteisessa avaruudessa määritellyn potentiaalienergiapinnan avulla. Energiapinnan paikalliset minimikohdat vastaavat systeemin vakaita tiloja, ja näitä tiloja yhdistävät minimienergiapolut kuvaavat mahdollisia reaktiomekanismeja. Erityisen mielenkiinnon kohteena on usein minimienergiapolun globaali maksimikohta, joka sijaitsee potentiaalienergiapinnan satulapisteessä ja jonka avulla voidaan arvioida kyseisen reaktion aktivoitumisenergiaa ja reaktionopeutta.

Kahden tunnetun tilan välisiä minimienergiapolkuja ja satulapisteitä on perinteisesti etsitty iteratiivisilla menetelmillä, joissa erillisistä koordinaattiavaruuden pisteistä muodostuvaa ketjua liikutetaan ja venytetään kohti minimienergiapolkua potentiaalienergiapinnan gradienttivektorien avulla laskettujen kuvitteellisten voimien perusteella. Varsinainen satulapiste voidaan määrittää kääntämällä gradienttivektorin polun suuntainen komponentti yhdessä ketjun pisteistä, jonka annetaan nousta polun suuntaisesti kohti satulapistettä. Jos reaktion lopputila on tuntematon, voidaan satulapistettä etsiä vastaavasti kiertämällä kahden lähekkäisen pisteen muodostamaa paria potentiaalienergiapinnan pienimmän kaarevuuden suuntaiseksi, kääntämällä tätä suuntaa vastaava gradienttikomponentti, ja liikuttamalla pisteparia kohti satulapistettä. Nämä meneltelmät voivat kuitenkin vaatia satoja iteraatioita, ja koska gradienttivektorin tarkka määrittäminen on usein laskennallisesti raskasta, tulisi aikaisemmista iteraatioista saatu informaatio hyödyntää mahdollisimman tehokkaasti iteraatioiden vähentämiseksi. Tilastollisten mallien avulla energiapinnalle voidaan muodostaa likimääräinen arvio, ja minimienergiapolkua tai satulapistettä voidaan etsiä likimääräiseltä pinnalta. Ratkaisu voidaan tarkistaa uusien tarkkojen havaintojen avulla, joita voidaan puolestaan käyttää mallin tarkentamiseksi mahdollisia seuraavia iteraatioita varten.

Tässä väitöskirjassa kehitetään gaussisiin prosesseihin perustuvia koneoppimisalgoritmeja minimienergiapolkujen ja satulapisteiden etsinnän nopeuttamiseksi. Gaussiset prosessit toimivat tässä tapauksessa joustavina prioritodennäköisyysmalleina potentiaalienergiapinnoille. Mallin päivittämiseksi voidaan käyttää sekä energian että gradienttikomponenttien havaittuja arvoja, ja Bayes-päättelyn tuloksena saatava ennustejakauma sisältää myös epävarmuusarvion, jota voidaan käyttää hyväksi uusien havaintopisteiden valinnassa. Väitöskirjassa esitetään menetelmät sekä kahden tunnetun tilan välisen minimienergiapolun määrittämiseen että annetun aloituspisteen lähistöllä sijaitsevan satulapisteen etsimiseen. Yksinkertaisten testiesimerkkien perusteella gaussisia prosesseja hyödyntävät menetelmät voivat vähentää tarkkojen havaintojen määrän murto-osaan perinteisten menetelmien vaatimista havaintomääristä.

Avainsanat satulapiste, minimienergiapolku, gaussinen prosessi, koneoppiminen

ISBN (painettu) 978-952-60-8850-1 **ISBN (pdf)** 978-952-60-8851-8

ISSN (painettu) 1799-4934 **ISSN (pdf)** 1799-4942

Julkaisupaikka Helsinki **Painopaikka** Helsinki **Vuosi** 2019

Sivumäärä 144 **urn** http://urn.fi/URN:ISBN:978-952-60-8851-8

Útdráttur

Eiginleika efnahvarfa og annarra umraðana atóma er hægt að kanna með því að skoða orkuyfirborðið, skilgreint sem orka kerfisins sem fall af atómhnitunum. Staðbundin lágmörk á orkuyfirborðinu samsvara ástöndum sem kerfið getur verið í og lágmarksorkuferlar milli þeirra einkenna gang mögulegra umraðana atómanna. Hámörk á lágmarksorkuferlum eru sérlega mikilvæg. Þau samsvara fyrsta stigs söðulpunktum og gefa mat á virkjunarorku og þar með hraða samsvarandi umröðunar.

Lágmarksorkuferlar og söðulpunktar milli tveggja þekktra ástanda eru gjarnan fundnir með ítrekunaraðferðum þar sem röð af ímyndum af kerfinu mynda feril milli endapunktanna og eru færðar til þangað til þær liggja á lágmarksorkuferlinum. Færslan í hverri ítrekun er fundin út frá stiglunum á orkuyfirborðinu. Söðulpunktinn er hægt að finna með því að færa orkuhæstu ímyndina í átt stigilsins eftir að þætti hans í stefnu ferilsins hefur verið snúið við. Þannig færist sú ímynd upp í orku að söðulpunktinum. Ef lokaástand umröðuninnar er ekki þekkt er hægt að finna fyrsta stigs söðulpunkt með því að nota par ímynda af kerfinu sem eru þétt saman og myndar tvennu. Henni er snúið til að finna stefnuna með lægstan krappa á orkuyfirborðinu og síðan færð í átt stigulsins eftir að þátturinn í stefnu lægsta krappans hefur verið speglaður. Þannig færist tvennan að söðulpunktinum. Þessi aðferð getur þurft hundruða ítrekana og þar eð útreikningar á orkustiglinum eru oft þungir er mikilvægt að nýta upplýsingar úr fyrri ítrekunum eins vel og hægt er til að fækka ítrekunum. Með því að nota tölfræðileg líkön er hægt að búa til nálgun fyrir orkuyfirborðið og leita að söðulpunktinum á því yfirborði. Lausnina er hægt að sanreyna með frekari útreikningum á orkustiglinum sem síðan er hægt að nota til að bæta nálgunina fyrir næstu færslur tvennunnar.

Í þessari ritgerð er vélrænn lærdómur sem byggður er á Gaussferlaaðhvarfi notaður til að hraða reikningum á lágmarksorkuferlum og söðulpunktum. Líkön fyrir orkuyfirborðið eru búin til með út frá þekktum gildum á orkunni og stiglinum með tölfræðilegum aðferðum Bayes og mat fundið á óvissunni í líkaninu sem hægt er að nýta til að ákveða hvaða punkt er best að reikna í næstu ítrekun. Mismunandi aðferðir eru þróaðar bæði til að finna lágmarksorkuferla milli tveggja þekktra ástanda og til að finna söðulpunkt í nágreni gefins upphafspunkts. Reikningar á ýmsum mismunandi kerfum sýna að með þessu móti er hægt að fækka útreikningum á orkunni og stiglinum mjög verulega í samanburði við þær aðferðir sem nú eru notaðar.

Contents

Preface

The journey towards this dissertation begun already in 2011 when I applied for a summer job at the Department of Biomedical Engineering and Computational Science (BECS) and found myself in the Bayesian methodology group, developing analysis methods for brain research. After finishing my master's thesis and pondering my future for a while, I eventually decided to continue on the track towards a doctoral degree. Along the journey, BECS became NBE, the Department of Neuroscience and Biomedical Engineering, and our group moved to the Department of Computer Science to be part of the current probabilistic machine learning group. These organizational changes naturally weakened our connection to neuroscience and spread the method development to a broader range of applications. After several, more or less successful trial projects, the final topic for my dissertation was quite unexpectedly found from the field of theoretical chemistry. This connection opened an interesting opportunity for a double degree via a joint supervision agreement between Aalto University and University of Iceland. During the time of the shared supervision, I was partly employed by the Department of Applied Physics. In addition to the employer departments, I gratefully acknowledge the financial support of the Academy of Finland and the Finnish Cultural Foundation (Kari Kairamo Fund) as well as the support of the Icelandic Research Fund to partly cover the expenses of my visits to Iceland.

Although research is at times lonely work inside one's own head, it is above all collaboration and learning from others. The people that I have learned of the most about science are my two supervisors, Prof. Aki Vehtari and Prof. Hannes Jónsson. As the leader of the former Bayesian methodology group, Aki has been supporting my work from the beginning and, by opening his bottomless storage of ideas, taken care that his students are never left empty-handed. On the other hand, I thank Aki also for the freedom he has given to develop the ideas further and patience when waiting for results. Hannes joined the journey in 2016 after he had met Aki at a conference and recognized a possibility for fruitful collaboration. That meeting turned out to be a good fortune to me as I got a well-defined goal for my theretofore unstructured doctoral research. I thank Hannes for warmly welcoming me to Iceland, introducing me to the necessary chemical details, and motivating me to find ways to tackle the methodological challenges. I want to express my gratitude also to Prof. Jouko Lampinen for the supervision in the initial phase of my doctoral studies, Prof. Johannes Kästner and Prof. Andrew Peterson for the pre-examination of this dissertation, and Prof. Thomas Bligaard for accepting the invitation to act as an opponent in the upcoming defence.

In addition to Aki and Hannes, I have the joy to acknowledge three more co-authors who have contributed to the articles included in the dissertation, Dr. Emile Maras, Freyja Dagbjartsdóttir, and Vilhjálmur Ásgeirsson. I thank Emile for introducing me to the details of the nudged elastic band method, Freyja for carefully performing the experiments for the heptamer island benchmark, and especially Villi for the close collaboration during the past three years. During the years at Aalto University, I have had a pleasure to work with many friendly and talented colleagues who have made my days easier through both work-related and more relaxed conversations and comments. I want to thank especially Akash Dhaka, Kunal Ghosh, Dr. Pasi Jylänki, Marko Järvenpää, Dr. Juho Kokkala, Sasu Mäkelä, Topi Paananen, Dr. Tomi Peltola, Dr. Juho Piironen, Prof. Michael Riis Andersen, Gabriel Riutort-Mayol, Prof. Juha Salmitaival, Eero Siivola, Tuomas Sivula, Dr. Dmitry Smirnov, and Prof. Arno Solin. I am grateful also to the service personnel at Aalto for the effort to secure the conditions for our research work.

Beside the studies, I have been lucky to have a passion towards the sport of orienteering. Although finding the balance between two demanding ambitions has been a challenge, orienteering has had an important role in keeping my thoughts away from research when needed. For the same reason, I want to thank all my friends, many of whom I have got to know in the activities of Teekkarisuunnistajat and Hiidenkiertäjät. A special mention is dedicated to Dr. Joonas Pääkkönen and Dr. Rainer Kujala, who have shared the same route choice with me also in studies and opened the way especially on the final legs of the course. Finally, and most importantly, I thank my family for all their support and encouragement. It is comforting to know that there are people you can always lean on, whatever comes about.

Espoo, December 11, 2019,

Olli-Pekka Koistinen

List of publications

This dissertation consists of an overview and of the following publications which are referred to in the text by their Roman numerals.

I Olli-Pekka Koistinen, Emile Maras, Aki Vehtari, and Hannes Jónsson. Minimum energy path calculations with Gaussian process regression. *Nanosystems: Physics, Chemistry, Mathematics*, volume 7, issue 6, pages 925–935, December 2016.

II Olli-Pekka Koistinen, Freyja B. Dagbjartsdóttir, Vilhjálmur Ásgeirsson, Aki Vehtari, and Hannes Jónsson. Nudged elastic band calculations accelerated with Gaussian process regression. *The Journal of Chemical Physics*, volume 147, issue 15, article 152720, 14 pages, September 2017.

III Olli-Pekka Koistinen, Vilhjálmur Ásgeirsson, Aki Vehtari, and Hannes Jónsson. Nudged elastic band calculations accelerated with Gaussian process regression based on inverse interatomic distances. *Journal of Chemical Theory and Computation*, volume 15, issue 12, pages 6738–6751, October 2019.

IV Olli-Pekka Koistinen, Vilhjálmur Ásgeirsson, Aki Vehtari, and Hannes Jónsson. Minimum mode saddle point searches using Gaussian process regression with inverse-distance covariance function. Accepted for publication in *Journal of Chemical Theory and Computation*, 20 pages, December 2019.

Author's contribution

Publication I: "Minimum energy path calculations with Gaussian process regression"

The initial idea of using Gaussian process regression with derivative observations in minimum energy path calculations came from Jónsson and Vehtari. Koistinen implemented and developed the GP-NEB algorithm, performed the experiments, and wrote parts of the manuscript describing Gaussian process methodology and details of the algorithm. Jónsson had the main responsibility in writing the manuscript. Vehtari contributed to the development of the algorithm and proposed suggestions to the manuscript regarding Gaussian process methodology. Maras advised in implementation of the NEB method and the experiments.

Publication II: "Nudged elastic band calculations accelerated with Gaussian process regression"

Koistinen innovated, implemented and developed the one-image-evaluated variant of the GP-NEB algorithm, performed initial experiments, and wrote parts of the manuscript describing Gaussian process methodology and details of the algorithm. Jónsson suggested extending the algorithm to climbing image NEB and including Hessian observations and had the main responsibility in writing the manuscript together with Koistinen. Vehtari supported the development of the algorithm and proposed suggestions to the manuscript regarding Gaussian process methodology. Dagbjartsdóttir performed the final experiments together with Ásgeirsson.

Publication III: "Nudged elastic band calculations accelerated with Gaussian process regression based on inverse interatomic distances"

Koistinen innovated, implemented and developed the methodological improvements to the GP-NEB algorithm, performed initial experiments and part of the final experiments, and wrote the manuscript. Jónsson supported the development of the algorithm and revised the manuscript. Vehtari supported the development of the algorithm, suggested various alternative improvements tested by Koistinen, and proposed suggestions to the manuscript regarding Gaussian process methodology. Ásgeirsson implemented the H_2/Cu(110) and H_2O potentials, performed part of the final experiments, and reviewed the manuscript.

Publication IV: "Minimum mode saddle point searches using Gaussian process regression with inverse-distance covariance function"

Koistinen implemented and developed the GP-dimer algorithm, performed the experiments, and wrote majority of the manuscript. Jónsson suggested the topic, supported the development of the algorithm, and wrote parts of the manuscript. Ásgeirsson implemented the potentials and reviewed the manuscript. Vehtari supported the development of the algorithm and reviewed the manuscript.

Abbreviations

AIE	all-images-evaluated
C	carbon
CI-NEB	climbing image nudged elastic band
Cu	copper
FCC	face-centred cubic
FIRE	fast inertial relaxation engine
GP	Gaussian process
GPR	Gaussian process regression
GPU	graphics processing unit
H	hydrogen
L-BFGS	limited-memory Broyden-Fletcher-Goldfarb-Shanno
N	nitrogen
NEB	nudged elastic band
O	oxygen
OIE	one-image-evaluated
S	sulphur
SOAP	smooth overlap of atomic positions
VPO	velocity projection optimization

Symbols

A_f	set of indices for frozen atoms
A_m	set of indices for moving atoms
$B_\nu(\cdot)$	modified Bessel function of the second kind and of order ν
C	curvature estimate
$\mathrm{Cov}[\cdot,\cdot]$	covariance
D	dimension
$\mathscr{D}_x(\cdot,\cdot)$	regular difference measure
$\mathscr{D}_{1/r}(\cdot,\cdot)$	inverse-distance difference measure
$E(\cdot)$	energy
$\mathrm{E}[\cdot]$	mean
$f(\cdot)$	latent function
$\mathbf{f}$	vector of latent function values at observation points
$\mathbf{f}^*$	vector of latent function values at prediction points
$\mathbf{F}(\cdot)$	atomic force vector (negative energy gradient)
$\mathbf{F}^\parallel(\cdot)$	parallel component of an atomic force vector
$\mathbf{F}^\perp(\cdot)$	perpendicular component of an atomic force vector
$\mathbf{F}_i^\mathrm{NEB}$	NEB force at the $(i+1)^\mathrm{th}$ image
$\mathbf{F}_i^\mathrm{s}$	spring force at the $(i+1)^\mathrm{th}$ image
$\mathbf{F}_\mathrm{rot}$	rotational force
$\mathbf{F}_\mathrm{trans}$	translational force
i_CI	index for the climbing image
$\mathbf{I}_N$	$N \times N$ identity matrix
$k(\cdot,\cdot)$	covariance function
k^s	spring constant
k_i^s	spring constant between i^th and $(i+1)^\mathrm{th}$ images

$k_x(\cdot,\cdot)$	squared exponential covariance function
$k_x^{\mathrm{M}-3/2}(\cdot,\cdot)$	Matérn-3/2 covariance function
$k_x^{\mathrm{M}-5/2}(\cdot,\cdot)$	Matérn-5/2 covariance function
$k_{1/r}(\cdot,\cdot)$	inverse-distance covariance function
$K(\cdot,\cdot)$	prior covariance matrix
$\mathbf{K}_{\mathrm{ext}}$	extended prior covariance matrix
$\mathbf{K}_{\mathrm{ext}}^*$	extended prior covariance matrix
$\mathbf{K}_{\mathbf{f}}$	posterior covariance matrix
$\mathbf{K}_{\mathbf{f}^*}$	posterior predictive covariance matrix
$\mathbf{K}_{\mathbf{f}^*\mid\mathbf{f}}$	conditional covariance matrix
l	length scale of a covariance function
l_d	length scale for the d^{th} input coordinate
$l_{\phi(\cdot,\cdot)}$	length scale for an atom pair
$m(\cdot)$	mean function
$\mathbf{m}$	prior mean vector
$\mathbf{m}_{\mathbf{f}}$	posterior mean vector
$\mathbf{m}_{\mathbf{f}^*}$	posterior predictive mean vector
$\mathbf{m}_{\mathbf{f}^*\mid\mathbf{f}}$	conditional mean vector
N	number of observation points
N^*	number of prediction points
N_{im}	number of images in a nudged elastic band
N_{m}	number of moving atoms
$\hat{\mathbf{N}}$	dimer orientation
$\hat{\mathbf{N}}^*$	dimer orientation after a preliminary rotation
$\mathcal{N}(\cdot,\cdot)$	Gaussian distribution
$p(\cdot)$	probability density
$r_{i,j}(\cdot)$	distance between atoms i and j
$\mathbf{R}_0$	middle point of a dimer
$\mathbf{R}_1$	image 1 of a dimer
$\mathbf{R}_1^*$	image 1 of a dimer after a preliminary rotation
$\mathbf{R}_2$	image 2 of a dimer
$\mathbf{R}_i$	the $(i+1)^{\mathrm{th}}$ image in a nudged elastic band
$\mathbb{R}$	the set of real numbers
$\mathrm{Var}[\cdot]$	variance

x_d	the d^{th} input coordinate
$x_{i,d}$	the d^{th} coordinate of atom i
$\mathbf{x}$	input vector
$\mathbf{x}^{(i)}$	the i^{th} observation point
$\mathbf{x}^{*(i)}$	the i^{th} prediction point
$\mathbf{X}$	matrix of observation points
$\mathbf{X}^*$	matrix of prediction points
$y^{(i)}$	the i^{th} output observation
$\mathbf{y}$	vector of output observations
$\mathbf{y}_{\mathrm{ext}}$	extended observation vector
$\mathbb{Z}$	the set of integers
$\Gamma(\cdot)$	gamma function
$\Delta_{\mathbf{R}}$	dimer separation
$\boldsymbol{\theta}$	vector of hyperparameters
$\boldsymbol{\theta}_{\mathrm{MAP}}$	maximum a posteriori estimate for hyperparameters
$\boldsymbol{\theta}_{\mathrm{ML}}$	maximum likelihood estimate for hyperparameters
ν	smoothness parameter of the Matérn covariance function
$\boldsymbol{\rho}$	vector of parameters
σ^2	noise variance
σ_{c}^2	constant covariance
σ_{d}^2	noise variance for derivatives
σ_{m}	magnitude of a covariance function
$\boldsymbol{\Sigma}$	extended noise covariance matrix
$\boldsymbol{\tau}_i$	unnormalized path tangent at the $(i+1)^{\mathrm{th}}$ image
$\hat{\boldsymbol{\tau}}_i$	normalized path tangent at the $(i+1)^{\mathrm{th}}$ image
$\phi(\cdot,\cdot)$	atom pair type
ω	rotation angle
ω^*	preliminary rotation angle
$\hat{\boldsymbol{\Omega}}$	rotation direction
$\hat{\boldsymbol{\Omega}}^*$	rotation direction after a preliminary rotation

1. Introduction

Theoretical chemistry utilizes physics, mathematics and computer science to explain and predict structural and dynamical properties of molecules and materials. One of the key concepts in theoretical chemistry is a potential energy surface, often described as a function in a high-dimensional space of atom coordinates, which contains the essential information of the properties of the system at a finite temperature. The most interesting locations on the energy surface are its local minimum points, corresponding to stable states of the system, and first-order saddle points located at energy ridges separating those states. Transitions from one state to another, caused by thermal fluctuations, can be characterized by a minimum energy path connecting the two states, and the highest point of this path is located at a first-order saddle point. The minimum energy path cannot be considered as an actual trajectory for the transition but rather a path of maximal statistical weight. In principle, such transitions could be simulated by classical dynamics, but since the time scale of the transition is often extremely large compared to the frequency of the thermal vibrations, statistical tools such as transition state theory (Wigner, 1938; Kramers, 1940; Keck, 1967) are required. A common approach is the harmonic approximation to the transition state theory (Vineyard, 1957), where the rate of the transition is estimated based on the energy and its second derivatives at the initial state and the saddle point.

Given an initial configuration of atoms, it is straightforward to locate the nearest minimum point on the energy surface with common optimization methods. A more challenging task is to find the saddle points located along the minimum energy paths leading to other relevant states of the system. A group of iterative algorithms, called surface walking methods or mode following methods, has been developed for the task to find a saddle point without knowledge of the final state of the transition. The common principle of these algorithms is to make the problem approachable for optimization methods by reverting the gradient component in the direction of the lowest energy curvature, i.e., the direction of the eigenvector corresponding to the lowest eigenvalue of the Hessian matrix, also known as the minimum curvature mode. With this modification, minimization of energy is supposed to lead to a first-order saddle point where the energy is maximized in the direction of the minimum energy path but minimized in all perpendicular directions. If the second derivatives of energy are easily available, all eigenvalues of the Hessian matrix can be calculated and a modified Hessian used to guide the saddle point search (Cerjan and Miller, 1981; Simons et al. 1983; Banerjee et al., 1985). A more efficient approach is to find the direction of the lowest curvature based only on the first derivatives (Henkelman and Jónsson, 1999; Munro and Wales, 1999; Malek and Mousseau, 2000). An example of such an approach is the dimer method (Henkelman and Jónsson, 1999), where a pair of closely spaced points is rotated towards the minimum curvature mode and translated towards a saddle point based on a modified gradient.

The task of finding a saddle point along a minimum energy path is easier, if also the final state of the transition has been found. In chain-of-states methods, such as the nudged elastic band (NEB) method (Mills et al., 1995; Jónsson et al., 1998), the path is represented as a discrete chain of points which is moved and stretched towards a minimum energy path so that the component of the energy gradient perpendicular to the path goes to zero at all points of the chain. In the NEB method, the distribution of the points along the path is controlled by a spring force acting parallel to the path. The actual saddle point can be found by reverting the gradient component parallel to the path at one of the points of the chain and letting this point climb along the path towards the saddle point.

Both surface walking and chain-of-states methods may require hundreds of iterations and evaluations of energy and its first derivatives. Since these evaluations typically involve computationally expensive electronic structure calculations, the information obtained from previous iterations should be utilized as efficiently as possible to decrease the number of iterations. A prominent approach for this purpose is to utilize machine learning to construct an approximate energy surface and perform the saddle point search based on the approximate model. The accuracy of the solution can be checked with further evaluations, which can then be used to update the model for the following iterations. Assuming that training of the machine learning model and evaluations of the approximate energy and derivatives are significantly cheaper than the accurate evaluations, the total number of the expensive evaluations can be reduced and the saddle point search hence accelerated. This general scheme has been introduced by Peterson (2016) with a demonstration of applying artificial neural network models to NEB calculations.

In this dissertation, similar algorithms to enhance searches of minimum energy paths and saddle points are developed using Gaussian process (GP) models as flexible prior probability models for potential energy surfaces. Observed values of both energy and its derivatives can be used to update the model, and the posterior predictive distribution obtained as a result of Bayesian inference provides also an uncertainty estimate, which can be utilized when selecting new observation points. Whereas optimization of a large number of weights of a neural network model may be challenging due to many local minima of the cost function, optimization of the hyperparameters of a GP model is typically a much easier task. Gaussian process regression have been shown to perform well especially when learning from small training data sets (Lampinen and Vehtari, 2001; Kamath et al., 2018), which makes it an appealing approach for this application. The GP-NEB algorithm (Publications I–III), based on the nudged elastic band method, finds a minimum energy path and a saddle point between two known states, whereas the GP-dimer algorithm (Publication IV), based on the dimer method, only finds a saddle point located in the vicinity of a given start point.

The dissertation consists of four articles and this overview part. The following chapter reviews the basics of Gaussian process regression, explains how to deal with derivatives in Gaussian process models, and shows how the framework is applied to modelling of potential energy surfaces in Publications I–IV. Chapter 3 reviews the regular nudged elastic band and dimer methods, and chapter 4 summarizes the contributions of Publications I–IV by explaining the main features of the GP-NEB and GP-dimer algorithms and presenting some test results.

2. Gaussian processes

Gaussian processes are a class of stochastic processes, particularly suitable for defining flexible prior distributions for functions in a Bayesian approach to supervised learning problems. Gaussian process models have been used for decades, e.g., in signal processing and geostatistics, where methods known as Wiener-Kolmogorov filtering (Kolmogorov, 1941; Wiener, 1949) and kriging (Krige, 1951; Matheron, 1963), respectively, correspond to Gaussian process regression. Bayesian interpretation of the GP framework has been presented by Kimeldorf and Wahba (1970), Blight and Ott (1975), and O'Hagan (1978) and later adopted by neural network researchers (Neal, 1995; Williams and Rasmussen, 1996; Rasmussen, 1996) who realized that neural network models in the limit of infinite number of hidden units can be handled elegantly by replacing the networks by Gaussian processes.

This chapter reviews the basics of Gaussian process regression from the Bayesian point of view, explains how to deal with derivatives in Gaussian process models, and finally shows how the framework is applied to approximation of potential energy surfaces in Publications I–IV. A more thorough review of the Bayesian approach to Gaussian process regression, including many of the basic equations appearing in this chapter, can be found in the book of Rasmussen and Williams (2006).

2.1 Gaussian process model

By definition, a Gaussian process is a collection of random variables with a multivariate Gaussian distribution for any finite set of these random variables. The random variables are most often indexed in a continuous domain such as time or space. In that case, the probability distribution of the Gaussian process itself is the infinite-dimensional joint distribution of all the random variables, in other words, a distribution over functions in a continuous input space. In the machine learning community, the term Gaussian process is often used to refer also to the model that defines the distribution of the process (see, e.g., Rasmussen and Williams, 2006). In this context, Gaussian processes compose a versatile modelling framework to specify prior probability distributions directly on functions and perform Bayesian inference on them based on observed data.

A Gaussian process model for the probability distribution of function $f : \mathbb{R}^D \rightarrow \mathbb{R}$ is specified by a mean function $m : \mathbb{R}^D \rightarrow \mathbb{R}$ and a covariance function $k : \mathbb{R}^D \times \mathbb{R}^D \rightarrow \mathbb{R}$. The mean function specifies the mean level of the marginal distribution of $f(\mathbf{x})$ at a given input point $\mathbf{x} \in \mathbb{R}^D$, i.e., $\mathrm{E}[f(\mathbf{x})] = m(\mathbf{x})$, and the covariance function specifies how the values of f at any two input points, $\mathbf{x}, \mathbf{x}' \in \mathbb{R}^D$, correlate with each other, more precisely, $\mathrm{E}[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}'))] = k(\mathbf{x}, \mathbf{x}')$. Given an arbitrary set of input points $\mathbf{X} = [\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(N)}]^\top$, the joint probability distribution of function values $\mathbf{f} = [f(\mathbf{x}^{(1)}), f(\mathbf{x}^{(2)}), \ldots, f(\mathbf{x}^{(N)})]^\top$ is defined as a multivariate Gaussian distribution

$$p(\mathbf{f}) = \mathcal{N}(\mathbf{m}, K(\mathbf{X}, \mathbf{X})) \tag{2.1}$$

with mean vector

$$\mathbf{m} = [m(\mathbf{x}^{(1)}), m(\mathbf{x}^{(2)}), \ldots, m(\mathbf{x}^{(N)})]^\top$$

and covariance matrix

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

2.2 Covariance functions

From now on, the mean function of the prior GP model is assumed to be set to zero, which is a common practice and applied also in Publications I–IV after a suitable shift of the zero level of the data. The essential part of a Gaussian process model is the covariance function, which can be used to encode favourable properties of the unknown function. From the perspective of machine learning, it has a particularly important role in defining what can be learned about the function based on observed values. If a covariance function $k(\mathbf{x}, \mathbf{x}')$ depends only on the vector between the two points, $\mathbf{x} - \mathbf{x}'$, it is called stationary since it behaves similarly in all parts of the input space. If a covariance function is also isotropic, it can be written simply as a function of the distance $||\mathbf{x} - \mathbf{x}'|| = \sqrt{\sum_{d=1}^{D} (x_d - x'_d)^2}$, which means that the behaviour is similar in all directions.

A common stationary example is the squared exponential (or perhaps more precisely exponentiated quadratic) covariance function

$$k(\mathbf{x}, \mathbf{x}') = \sigma_{\mathrm{m}}^2 \exp\left(-\frac{||\mathbf{x} - \mathbf{x}'||^2}{2l^2} \right) \tag{2.2}$$

where σ_m and l are the hyperparameters of the covariance function. The covariance is larger when the two input points are closer to each other and decreases with increasing distance. The magnitude σ_m defines the process variance, i.e., how much the values of f tend to deviate from the mean function, and the length scale l defines how far the effect of the covariance function fades out. In this isotropic form, the length scale is the same in all directions, but it is also possible to give separate length scales l_d for each input coordinate $d = 1, \ldots, D$:

$$k(\mathbf{x},\mathbf{x}') = \sigma_\mathrm{m}^2 \exp\left(-\sum_{d=1}^{D} \frac{\left(x_d - x'_d\right)^2}{2l_d^2} \right). \tag{2.3}$$

A GP model with a squared exponential covariance function favours extremely smooth functions. This property stems from the fact that the covariance function is infinite times differentiable, implying that sample functions drawn from the probability model are as well infinite times differentiable.

Even though the squared exponential covariance function is one of the most popular choices for a GP model, such a demanding smoothness assumption may be unrealistic for some real-world applications (Stein, 1999). The Matérn class of covariance functions (Matérn, 1960) allows to loosen the smoothness assumptions by adjusting an additional hyperparameter ν. The general form of the isotropic Matérn covariance function is given by

$$k(\mathbf{x},\mathbf{x}') = \sigma_\mathrm{m}^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu}||\mathbf{x}-\mathbf{x}'||}{l} \right)^{\nu} B_\nu \left(\frac{\sqrt{2\nu}||\mathbf{x}-\mathbf{x}'||}{l} \right), \tag{2.4}$$

where Γ denotes the gamma function and B_ν the modified Bessel function of the second kind (Olver and Maximon, 2010). A more convenient presentation is obtained when $\nu = p + 1/2$, where $0 \leq p \in \mathbb{Z}$:

$$k(\mathbf{x},\mathbf{x}') = \sigma_\mathrm{m}^2 \exp\left(-\frac{\sqrt{2\nu}||\mathbf{x}-\mathbf{x}'||}{l} \right) \frac{p!}{(2p)!} \sum_{i=0}^{p} \left(\frac{(p+i)!}{i!(p-i)!} \left(\frac{2\sqrt{2\nu}||\mathbf{x}-\mathbf{x}'||}{l} \right)^{p-i} \right).$$

Sample functions drawn from this model are n times differentiable when $n > \nu$. When ν approaches infinity, the Matérn class converges to the squared exponential covariance function, whereas a choice of $\nu = 1/2$ leads to the exponential covariance function

$$k(\mathbf{x},\mathbf{x}') = \sigma_\mathrm{m}^2 \exp\left(-\frac{||\mathbf{x}-\mathbf{x}'||}{l} \right) \tag{2.5}$$

and continuous but non-differentiable, roughly varying sample functions. In practice, a good compromise for the smoothness assumption is often obtained by choosing a once differentiable process with $\nu = 3/2$, so that

$$k(\mathbf{x},\mathbf{x}') = \sigma_\mathrm{m}^2 \left(1 + \frac{\sqrt{3}||\mathbf{x}-\mathbf{x}'||}{l} \right) \exp\left(-\frac{\sqrt{3}||\mathbf{x}-\mathbf{x}'||}{l} \right), \tag{2.6}$$

or a twice differentiable process with $\nu = 5/2$, so that

$$k(\mathbf{x},\mathbf{x}') = \sigma_{\mathrm{m}}^2\left(1 + \frac{\sqrt{5}||\mathbf{x}-\mathbf{x}'||}{l} + \frac{5||\mathbf{x}-\mathbf{x}'||^2}{3l^2}\right)\exp\left(-\frac{\sqrt{5}||\mathbf{x}-\mathbf{x}'||}{l}\right). \tag{2.7}$$

Similarly as for the squared exponential covariance function, it is possible to give separate length scales l_d for each input coordinate $d = 1,\ldots,D$ by replacing the scaled distance $||\mathbf{x}-\mathbf{x}'||/l$ with $\sqrt{\sum_{d=1}^{D}((x_d - x_d')/l_d)^2}$.

One more simple covariance function encountered in this dissertation is the constant function

$$k(\mathbf{x},\mathbf{x}') = \sigma_{\mathrm{c}}^2, \tag{2.8}$$

often used as an auxiliary term with other covariance functions. As a constant covariance implies full correlation between all function values, adding σ_{c}^2 to the covariance function corresponds to adding a constant intercept term to the process so that the unknown constant has a Gaussian prior distribution with variance σ_{c}^2. Thus, the constant covariance term can be used to allow variation of the global mean level even if the mean function was set to zero.

2.3 Gaussian process regression

Consider a regression problem with a training data set $\{\mathbf{X},\mathbf{y}\}$, including output observations $\mathbf{y} = [y^{(1)}, y^{(2)}, \ldots, y^{(N)}]^\top$ made at N input points $\mathbf{X} = [\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(N)}]^\top$, and an observation model

$$p(\mathbf{y}|\mathbf{f}) = \prod_{i=1}^{N} p\big(y^{(i)}\big|f(\mathbf{x}^{(i)})\big), \tag{2.9}$$

where $\mathbf{f} = [f(\mathbf{x}^{(1)}), f(\mathbf{x}^{(2)}), \ldots, f(\mathbf{x}^{(N)})]^\top$ is a vector of latent function values at the input data points. In a typical Bayesian modelling approach, the latent function $f(\mathbf{x})$ would be specified by a set of unknown parameters $\boldsymbol{\rho}$ with a prior distribution $p(\boldsymbol{\rho})$. According to the Bayes' theorem, the posterior distribution of $\boldsymbol{\rho}$ conditioned on the training data would be given by

$$p(\boldsymbol{\rho}\,|\,\mathbf{X},\mathbf{y}) = \frac{p(\mathbf{y}\,|\,\mathbf{X},\boldsymbol{\rho})p(\boldsymbol{\rho})}{p(\mathbf{y}\,|\,\mathbf{X})}, \tag{2.10}$$

where $p(\mathbf{y}\,|\,\mathbf{X},\boldsymbol{\rho})$ with fixed $\mathbf{y}$ and $\mathbf{X}$ is the likelihood of $\boldsymbol{\rho}$ given by the observation model and the normalization constant $p(\mathbf{y}\,|\,\mathbf{X})$ is obtained by integrating over the parameters,

$$p(\mathbf{y}|\mathbf{X}) = \int_{\boldsymbol{\rho}} p(\mathbf{y}\,|\,\mathbf{X},\boldsymbol{\rho})p(\boldsymbol{\rho})\,\mathrm{d}\boldsymbol{\rho}. \tag{2.11}$$

In Gaussian process regression, the prior distribution is given directly to the values of the latent function f. For this reason, Gaussian process models are often called non-parametric, but sometimes also infinite-parametric since the unlimited collection of latent function values $\mathbf{f}$ at the training input points can be seen as the parameters of the model. When modelling the prior of f with a Gaussian process with mean function $m(\mathbf{x}) = \mathbf{0}$ and a prior covariance function $k(\mathbf{x}, \mathbf{x}' \mid \boldsymbol{\theta})$, the posterior distribution of $\mathbf{f}$, conditional on the hyperparameters $\boldsymbol{\theta}$ of the covariance function, is given by

$$p(\mathbf{f} \mid \mathbf{X}, \mathbf{y}, \boldsymbol{\theta}) = \frac{p(\mathbf{y} \mid \mathbf{f}) p(\mathbf{f} \mid \mathbf{X}, \boldsymbol{\theta})}{p(\mathbf{y} \mid \mathbf{X}, \boldsymbol{\theta})}. \tag{2.12}$$

Generally, evaluation of this distribution requires approximative methods such as Monte Carlo integration (Neal, 1999), Laplace approximation (Williams and Barber, 1998), expectation propagation (Minka, 2001), or variational methods (Gibbs and MacKay, 2000), but in case of a Gaussian observation model

$$p(\mathbf{y} \mid \mathbf{f}) = \prod_{i=1}^{N} \mathcal{N}\left(y^{(i)} \middle| f(\mathbf{x}^{(i)}), \sigma^2\right), \tag{2.13}$$

the posterior can be presented in an analytical Gaussian form:

$$p(\mathbf{f} \mid \mathbf{X}, \mathbf{y}, \boldsymbol{\theta}) = \mathcal{N}(\mathbf{f} \mid \mathbf{m}_{\mathbf{f}}, \mathbf{K}_{\mathbf{f}}), \tag{2.14}$$

where

$$\mathbf{m}_{\mathbf{f}} = K(\mathbf{X}, \mathbf{X}) \left(K(\mathbf{X}, \mathbf{X}) + \sigma^2 \mathbf{I}_N \right)^{-1} \mathbf{y}$$

and

$$\mathbf{K}_{\mathbf{f}} = K(\mathbf{X}, \mathbf{X}) - K(\mathbf{X}, \mathbf{X}) \left(K(\mathbf{X}, \mathbf{X}) + \sigma^2 \mathbf{I}_N \right)^{-1} K(\mathbf{X}, \mathbf{X})$$

with $\mathbf{I}_N$ denoting an $N \times N$ identity matrix.

To predict function values $\mathbf{f}^* = [f(\mathbf{x}^{*(1)}), f(\mathbf{x}^{*(2)}), \ldots, f(\mathbf{x}^{*(N^*)})]^\top$ at an arbitrary set of input points $\mathbf{X}^* = [\mathbf{x}^{*(1)}, \mathbf{x}^{*(2)}, \ldots, \mathbf{x}^{*(N^*)}]^\top$, consider first the joint prior distribution of $\mathbf{f}$ and $\mathbf{f}^*$:

$$p\left(\begin{bmatrix} \mathbf{f} \\ \mathbf{f}^* \end{bmatrix} \middle| \begin{bmatrix} \mathbf{X} \\ \mathbf{X}^* \end{bmatrix} \right) = \mathcal{N}\left(\begin{bmatrix} \mathbf{f} \\ \mathbf{f}^* \end{bmatrix} \middle| \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} K(\mathbf{X}, \mathbf{X}) & K(\mathbf{X}, \mathbf{X}^*) \\ K(\mathbf{X}^*, \mathbf{X}) & K(\mathbf{X}^*, \mathbf{X}^*) \end{bmatrix} \right). \tag{2.15}$$

According to the conditionalization properties of the multivariate Gaussian distribution (Mises, 1964), the conditional distribution of $\mathbf{f}^*$, given $\mathbf{f}$, becomes

$$p(\mathbf{f}^* \mid \mathbf{X}^*, \mathbf{X}, \mathbf{f}, \boldsymbol{\theta}) = \mathcal{N}\left(\mathbf{f}^* \middle| \mathbf{m}_{\mathbf{f}^*|\mathbf{f}}, \mathbf{K}_{\mathbf{f}^*|\mathbf{f}}\right), \tag{2.16}$$

where

$$\mathbf{m}_{\mathbf{f}^*|\mathbf{f}} = K(\mathbf{X}^*, \mathbf{X}) K(\mathbf{X}, \mathbf{X})^{-1} \mathbf{f}$$

and

$$\mathbf{K}_{\mathbf{f}^*|\mathbf{f}} = K(\mathbf{X}^*, \mathbf{X}^*) - K(\mathbf{X}^*, \mathbf{X}) K(\mathbf{X}, \mathbf{X})^{-1} K(\mathbf{X}, \mathbf{X}^*).$$

The posterior predictive distribution for the function values $\mathbf{f}^*$ is obtained by marginalizing from the joint posterior

$$p(\mathbf{f}, \mathbf{f}^* \mid \mathbf{X}^*, \mathbf{X}, \mathbf{y}, \boldsymbol{\theta}) = p(\mathbf{f}^* \mid \mathbf{X}^*, \mathbf{X}, \mathbf{f}, \boldsymbol{\theta}) p(\mathbf{f} \mid \mathbf{X}, \mathbf{y}, \boldsymbol{\theta}).$$

With the Gaussian observation model, also this distribution remains Gaussian:

$$p(\mathbf{f}^*\,|\,\mathbf{X}^*,\mathbf{X},\mathbf{y},\boldsymbol{\theta}) = \int_{\mathbf{f}} p(\mathbf{f}^*\,|\,\mathbf{X}^*,\mathbf{X},\mathbf{f},\boldsymbol{\theta})p(\mathbf{f}|\mathbf{X},\mathbf{y},\boldsymbol{\theta})\,\mathrm{d}\mathbf{f} = \mathcal{N}\big(\mathbf{f}^*\,\big|\,\mathbf{m}_{\mathbf{f}^*},\mathbf{K}_{\mathbf{f}^*}\big), \tag{2.17}$$

where

$$\mathbf{m}_{\mathbf{f}^*} = K(\mathbf{X}^*,\mathbf{X})\big(K(\mathbf{X},\mathbf{X})+\sigma^2\mathbf{I}_N\big)^{-1}\mathbf{y}$$

and

$$\mathbf{K}_{\mathbf{f}^*} = K(\mathbf{X}^*,\mathbf{X}^*) - K(\mathbf{X}^*,\mathbf{X})\big(K(\mathbf{X},\mathbf{X})+\sigma^2\mathbf{I}_N\big)^{-1}K(\mathbf{X},\mathbf{X}^*).$$

The equations above are all conditional on the prior covariance function $k(\mathbf{x},\mathbf{x}')$ with known hyperparameters $\boldsymbol{\theta}$. The standard way to learn the hyperparameters is to maximize the marginal likelihood of $\boldsymbol{\theta}$ given a data set $\{\mathbf{X},\mathbf{y}\}$, appearing in the denominator of equation 2.12:

$$\boldsymbol{\theta}_{\mathrm{ML}} = \arg\max_{\boldsymbol{\theta}} p(\mathbf{y}\,|\,\mathbf{X},\boldsymbol{\theta}) = \arg\max_{\boldsymbol{\theta}} \int_{\mathbf{f}} p(\mathbf{y}\,|\,\mathbf{f})p(\mathbf{f}|\mathbf{X},\boldsymbol{\theta})\,\mathrm{d}\mathbf{f}. \tag{2.18}$$

With the Gaussian observation model, the marginal likelihood is simply given by

$$p(\mathbf{y}\,|\,\mathbf{X},\boldsymbol{\theta}) = \mathcal{N}\big(\mathbf{y}\,\big|\,\mathbf{0},K(\mathbf{X},\mathbf{X})+\sigma^2\mathbf{I}_N\big). \tag{2.19}$$

To improve stability and data efficiency, it is also possible to define a prior distribution $p(\boldsymbol{\theta})$ (hyperprior), as done in Publications I–IV, and maximize the marginal posterior probability density $p(\boldsymbol{\theta}\,|\,\mathbf{y},\mathbf{X}) \propto p(\boldsymbol{\theta})p(\mathbf{y}\,|\,\mathbf{X},\boldsymbol{\theta})$:

$$\boldsymbol{\theta}_{\mathrm{MAP}} = \arg\max_{\boldsymbol{\theta}} p(\boldsymbol{\theta}\,|\,\mathbf{y},\mathbf{X}) = \arg\max_{\boldsymbol{\theta}} p(\boldsymbol{\theta})p(\mathbf{y}\,|\,\mathbf{X},\boldsymbol{\theta}). \tag{2.20}$$

An alternative to a maximum a posteriori estimate would be to integrate over the uncertainty of the marginal posterior $p(\boldsymbol{\theta}\,|\,\mathbf{y},\mathbf{X})$ using approximations based on, e.g., Monte Carlo or grid sampling or a central composite design (Rue et al., 2009; Vanhatalo et al., 2010). In addition to the hyperparameters of the covariance function, the parameters of the observation model such as the noise variance σ^2 can be similarly treated as unknown hyperparameters and incorporated in the optimization or integration.

The elegance of Gaussian process regression relies on the implicit encoding of the function properties via selection of the covariance function, which allows flexible models without restricting to simple parametric forms. The strength of the framework is most apparent in prediction based on small training data sets (Lampinen and Vehtari, 2001; Kamath et al., 2018). The price of the elegance, however, is realized as computational challenges with large data sets, since training of the model involves solving a linear system associated with the training covariance matrix. This is typically performed via a Cholesky decomposition with a cubic computational cost with respect to the number of training observations, which makes large data sets infeasible (Rasmussen and Williams, 2006). Common ways to alleviate the problem involve compactly supported covariance functions leading to sparse covariance matrices with zero covariance between data points far away from each other (Sansò and Schuh, 1987; Wu, 1995; Wendland, 1995; Vanhatalo and Vehtari, 2008), sparse approximations by representing the training data set with a smaller set of inducing points (Csató and Opper, 2002; Seeger et al., 2003; Quiñonero-Candela and Rasmussen, 2005; Snelson and Ghahramani, 2006; Titsias, 2009; Hensman et al., 2013), and mixture-of-experts models where the computation can be distributed over several local data sets (Deisenroth and Ng, 2015). However, a more recent inference approach (Gardner et al., 2018; Wang et al., 2019) avoids the Cholesky decomposition by using a modified batched conjugate gradients algorithm and allows quadratic scaling without compromising the accuracy of the inference. In this approach, the covariance matrix is accessed through matrix-matrix multiplications which can be computed efficiently with GPU (graphics processing unit) hardware.

2.4 Regression with derivatives

In many applications, it is desirable to predict also the derivatives of f or incorporate information about the derivatives into the model. For Gaussian process models with differentiable covariance functions, this turns out to be straightforward since the linearity of differentiation implies that the derivative of a Gaussian process is another Gaussian process (O'Hagan, 1992; Rasmussen, 2003; Solak et al., 2003; Riihimäki and Vehtari, 2010). The covariance between a partial derivative at $\mathbf{x}$ and a function value at $\mathbf{x}'$ is simply given by differentiating the covariance function,

$$\mathrm{Cov}\left[\frac{\partial f(\mathbf{x})}{\partial x_d}, f(\mathbf{x}')\right] = \frac{\partial}{\partial_1 x_d}\mathrm{Cov}\left[f(\mathbf{x}), f(\mathbf{x}')\right] = \frac{\partial k(\mathbf{x},\mathbf{x}')}{\partial_1 x_d}, \tag{2.21}$$

and similarly,

$$\mathrm{Cov}\left[\frac{\partial f(\mathbf{x})}{\partial x_{d_1}}, \frac{\partial f(\mathbf{x}')}{\partial x'_{d_2}}\right] = \frac{\partial^2}{\partial_1 x_{d_1}\partial_2 x'_{d_2}}\mathrm{Cov}\left[f(\mathbf{x}), f(\mathbf{x}')\right] = \frac{\partial^2 k(\mathbf{x},\mathbf{x}')}{\partial_1 x_{d_1}\partial_2 x'_{d_2}}. \tag{2.22}$$

The notation ∂_1 indicates here that the covariance is differentiated with respect to a component of the first argument $\mathbf{x}$, and ∂_2 correspondingly refers to the second argument $\mathbf{x}'$.

To predict partial derivatives of f, vector $\mathbf{f}^*$ and covariance matrices $K(\mathbf{X}^*,\mathbf{X}^*)$ and $K(\mathbf{X}^*,\mathbf{X})$ in equations 2.15–2.17 can be extended as

$$\begin{bmatrix} \mathbf{f}^* \\ \frac{\partial f(\mathbf{X}^*)}{\partial x_1^*} \\ \frac{\partial f(\mathbf{X}^*)}{\partial x_2^*} \\ \vdots \\ \frac{\partial f(\mathbf{X}^*)}{\partial x_D^*} \end{bmatrix}, \begin{bmatrix} K(\mathbf{X}^*,\mathbf{X}^*) & \frac{\partial K(\mathbf{X}^*,\mathbf{X}^*)}{\partial_2 x_1^*} & \frac{\partial K(\mathbf{X}^*,\mathbf{X}^*)}{\partial_2 x_2^*} & \cdots & \frac{\partial K(\mathbf{X}^*,\mathbf{X}^*)}{\partial_2 x_D^*} \\ \frac{\partial K(\mathbf{X}^*,\mathbf{X}^*)}{\partial_1 x_1^*} & \frac{\partial K(\mathbf{X}^*,\mathbf{X}^*)}{\partial_1 x_1^*\partial_2 x_1^*} & \frac{\partial K(\mathbf{X}^*,\mathbf{X}^*)}{\partial_1 x_1^*\partial_2 x_2^*} & \cdots & \frac{\partial K(\mathbf{X}^*,\mathbf{X}^*)}{\partial_1 x_1^*\partial_2 x_D^*} \\ \frac{\partial K(\mathbf{X}^*,\mathbf{X}^*)}{\partial_1 x_2^*} & \frac{\partial K(\mathbf{X}^*,\mathbf{X}^*)}{\partial_1 x_2^*\partial_2 x_1^*} & \frac{\partial K(\mathbf{X}^*,\mathbf{X}^*)}{\partial_1 x_2^*\partial_2 x_2^*} & \cdots & \frac{\partial K(\mathbf{X}^*,\mathbf{X}^*)}{\partial_1 x_2^*\partial_2 x_D^*} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial K(\mathbf{X}^*,\mathbf{X}^*)}{\partial_1 x_D^*} & \frac{\partial K(\mathbf{X}^*,\mathbf{X}^*)}{\partial_1 x_D^*\partial_2 x_1^*} & \frac{\partial K(\mathbf{X}^*,\mathbf{X}^*)}{\partial_1 x_D^*\partial_2 x_2^*} & \cdots & \frac{\partial K(\mathbf{X}^*,\mathbf{X}^*)}{\partial_1 x_D^*\partial_2 x_D^*} \end{bmatrix}, \text{and} \begin{bmatrix} K(\mathbf{X}^*,\mathbf{X}) \\ \frac{\partial K(\mathbf{X}^*,\mathbf{X})}{\partial_1 x_1^*} \\ \frac{\partial K(\mathbf{X}^*,\mathbf{X})}{\partial_1 x_2^*} \\ \vdots \\ \frac{\partial K(\mathbf{X}^*,\mathbf{X})}{\partial_1 x_D^*} \end{bmatrix},$$

respectively. Often the primary interest is in the marginal posterior predictive distribution of individual variables, whereupon the covariances between predictions of different partial derivatives and between predictions at different input points can be ignored. For example, the posterior predictive distribution of the partial derivative of f with respect to input coordinate d at $\mathbf{x}^*$, assuming the Gaussian observation model, is a Gaussian distribution with mean

$$\mathrm{E}\left[\frac{\partial f(\mathbf{x}^*)}{\partial x_d^*}\,\middle|\,\mathbf{X},\mathbf{y},\boldsymbol{\theta}\right] = \frac{\partial K(\mathbf{x}^*,\mathbf{X})}{\partial x_d^*}\left(K(\mathbf{X},\mathbf{X})+\sigma^2\mathbf{I}_N\right)^{-1}\mathbf{y} \tag{2.23}$$

and variance

$$\mathrm{Var}\left[\frac{\partial f(\mathbf{x}^*)}{\partial x_d^*}\,\middle|\,\mathbf{X},\mathbf{y},\boldsymbol{\theta}\right] = \frac{\partial^2 k(\mathbf{x}^*,\mathbf{x}^*)}{\partial_1 x_d^*\partial_2 x_d^*} - \frac{\partial K(\mathbf{x}^*,\mathbf{X})}{\partial x_d^*}\left(K(\mathbf{X},\mathbf{X})+\sigma^2\mathbf{I}_N\right)^{-1}\frac{\partial K(\mathbf{X},\mathbf{x}^*)}{\partial x_d^*}. \tag{2.24}$$

Similarly, derivative observations can be included in the model by extending the observation vector $\mathbf{y}$ to include partial derivative observations and by extending the covariance matrices correspondingly. Assuming a Gaussian noise model also for the derivative observations, the posterior predictive mean and variance for f at $\mathbf{x}^*$ are then given as

$$\mathrm{E}\left[f(\mathbf{x}^*)\,\middle|\,\mathbf{y}_{\mathrm{ext}},\mathbf{X},\boldsymbol{\theta}\right] = \mathbf{K}_{\mathrm{ext}}^*(\mathbf{K}_{\mathrm{ext}}+\boldsymbol{\Sigma})^{-1}\mathbf{y}_{\mathrm{ext}} \tag{2.25}$$

and

$$\mathrm{Var}\left[f(\mathbf{x}^*)\,\middle|\,\mathbf{y}_{\mathrm{ext}},\mathbf{X},\boldsymbol{\theta}\right] = k(\mathbf{x}^*,\mathbf{x}^*) - \mathbf{K}_{\mathrm{ext}}^*(\mathbf{K}_{\mathrm{ext}}+\boldsymbol{\Sigma})^{-1}{\mathbf{K}_{\mathrm{ext}}^*}^{\mathsf{T}}, \tag{2.26}$$

where

$$\mathbf{y}_{\mathrm{ext}} = \begin{bmatrix} \mathbf{y} \\ \frac{\partial f(\mathbf{X})}{\partial x_1} \\ \frac{\partial f(\mathbf{X})}{\partial x_2} \\ \vdots \\ \frac{\partial f(\mathbf{X})}{\partial x_D} \end{bmatrix},\ \mathbf{K}_{\mathrm{ext}} = \begin{bmatrix} K(\mathbf{X},\mathbf{X}) & \frac{\partial K(\mathbf{X},\mathbf{X})}{\partial_2 x_1} & \frac{\partial K(\mathbf{X},\mathbf{X})}{\partial_2 x_2} & \cdots & \frac{\partial K(\mathbf{X},\mathbf{X})}{\partial_2 x_D} \\ \frac{\partial K(\mathbf{X},\mathbf{X})}{\partial_1 x_1} & \frac{\partial^2 K(\mathbf{X},\mathbf{X})}{\partial_1 x_1 \partial_2 x_1} & \frac{\partial^2 K(\mathbf{X},\mathbf{X})}{\partial_1 x_1 \partial_2 x_2} & \cdots & \frac{\partial^2 K(\mathbf{X},\mathbf{X})}{\partial_1 x_1 \partial_2 x_D} \\ \frac{\partial K(\mathbf{X},\mathbf{X})}{\partial_1 x_2} & \frac{\partial^2 K(\mathbf{X},\mathbf{X})}{\partial_1 x_2 \partial_2 x_1} & \frac{\partial^2 K(\mathbf{X},\mathbf{X})}{\partial_1 x_2 \partial_2 x_2} & \cdots & \frac{\partial^2 K(\mathbf{X},\mathbf{X})}{\partial_1 x_2 \partial_2 x_D} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial K(\mathbf{X},\mathbf{X})}{\partial_1 x_D} & \frac{\partial^2 K(\mathbf{X},\mathbf{X})}{\partial_1 x_D \partial_2 x_1} & \frac{\partial^2 K(\mathbf{X},\mathbf{X})}{\partial_1 x_D \partial_2 x_2} & \cdots & \frac{\partial^2 K(\mathbf{X},\mathbf{X})}{\partial_1 x_D \partial_2 x_D} \end{bmatrix},$$

$$\mathbf{K}_{\mathrm{ext}}^* = \begin{bmatrix} K(\mathbf{x}^*,\mathbf{X}) & \frac{\partial K(\mathbf{x}^*,\mathbf{X})}{\partial_2 x_1} & \frac{\partial K(\mathbf{x}^*,\mathbf{X})}{\partial_2 x_2} & \cdots & \frac{\partial K(\mathbf{x}^*,\mathbf{X})}{\partial_2 x_D} \end{bmatrix},$$

and

$$\boldsymbol{\Sigma} = \begin{bmatrix} \sigma^2\mathbf{I}_N & \mathbf{0} \\ \mathbf{0} & \sigma_{\mathrm{d}}^2\mathbf{I}_{ND} \end{bmatrix}$$

is the extended noise covariance matrix with noise variance σ_{d}^2 for the derivative observations. Correspondingly, the mean and variance of the posterior predictive distribution of the partial derivative of f with respect to coordinate d at $\mathbf{x}^*$ are given as

$$\mathrm{E}\left[\frac{\partial f(\mathbf{x}^*)}{\partial x_d^*}\,\middle|\,\mathbf{y}_{\mathrm{ext}},\mathbf{X},\boldsymbol{\theta}\right] = \frac{\partial \mathbf{K}_{\mathrm{ext}}^*}{\partial x_d^*}(\mathbf{K}_{\mathrm{ext}}+\boldsymbol{\Sigma})^{-1}\mathbf{y}_{\mathrm{ext}} \tag{2.27}$$

and

$$\mathrm{Var}\left[\frac{\partial f(\mathbf{x}^*)}{\partial x_d^*}\middle|\mathbf{y}_{\mathrm{ext}},\mathbf{X},\boldsymbol{\theta}\right]=\frac{\partial^2 k(\mathbf{x}^*,\mathbf{x}^*)}{\partial_1 x_d^* \partial_2 x_d^*}-\frac{\partial \mathbf{K}_{\mathrm{ext}}^*}{\partial_1 x_d^*}(\mathbf{K}_{\mathrm{ext}}+\boldsymbol{\Sigma})^{-1}\frac{\partial \mathbf{K}_{\mathrm{ext}}^*}{\partial_1 x_d^*}^{\mathsf{T}}. \tag{2.28}$$

Equation 2.27 is the central result used in Publications I–IV when predicting gradients of a potential energy surface based on a training data set including derivative observations.

2.5 Gaussian process models for potential energy surfaces

In this dissertation, Gaussian processes are used to model parts of potential energy surfaces in order to accelerate algorithms that aim to find minimum energy paths and saddle points on the energy surfaces. Following the notation of Publication III,

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

represents now a $3N_{\mathrm{m}}$-dimensional configuration vector including coordinates for moving atoms $1,2,\ldots,N_{\mathrm{m}} \in A_{\mathrm{m}}$ and f is the unknown energy of the system as a function of $\mathbf{x}$. The training data set consists of both the energy and its first derivatives with respect to the components of $\mathbf{x}$. The observations are here regarded as accurate up to floating point presentation accuracy, and thus only a really small Gaussian noise term is included in the observation model to avoid numerical issues. An approximation to the energy surface is given by the mean of the posterior predictive distribution of f (equation 2.25), and the variance of the distribution (equation 2.26) can be used as an uncertainty estimate for the GP approximation. As the algorithm proceeds, more observations are made and the model is updated until it is accurate enough to allow convergence to a minimum energy path and/or a saddle point.

In Publications I and II, a simple model with a stationary squared exponential covariance function k_x is successfully applied to meet the goals of the algorithms in a benchmark case involving rearrangements of a heptamer island on a crystal surface (Henkelman, Jóhannesson, and Jónsson, 2000; Chill et al., 2014):

$$k_x(\mathbf{x},\mathbf{x}')=\sigma_{\mathrm{c}}^2+\sigma_{\mathrm{m}}^2\exp\left(-\frac{1}{2}\mathscr{D}_x^2(\mathbf{x},\mathbf{x}')\right), \tag{2.29}$$

where

$$\mathscr{D}_x(\mathbf{x},\mathbf{x}')=\sqrt{\sum_{i=1}^{N_{\mathrm{m}}}\sum_{d=1}^{3}\frac{\left(x_{i,d}-x'_{i,d}\right)^2}{l^2}} \tag{2.30}$$

is a difference measure defined as a regular Euclidean distance between configuration vectors in the $3N_{\mathrm{m}}$-dimensional space of atom coordinates.

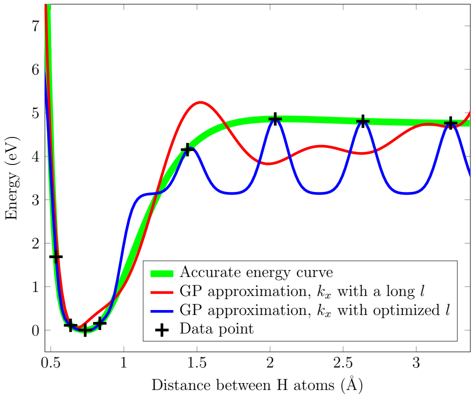


Figure 2.1. Illustration of problems in fitting a one-dimensional energy curve of a hydrogen molecule when using a Gaussian process model with stationary squared exponential covariance function k_x (equation 2.29). The training data points include accurate values for both energy and its first derivative. When the length scale of the covariance function is too long, the dominating data from the steep parts of the curve disturb the predictions at longer distances where the GP approximation does not match with the training data points even if the noise variance is assumed to be really small. When optimized, however, the length scale becomes too short for interpolation of the data points at the flat parts of the curve. Figure reproduced with permission from Publication III.

In some systems, however, strong and quickly changing repulsive forces may cause problems for stationary covariance functions, as demonstrated in Publication III. Figure 2.1 shows a simple example involving a pair of hydrogen atoms, where fitting a one-dimensional energy curve turns out to be a challenging task for covariance function k_x. With a too long length scale l, the dominating data from the steep part of the curve disturb the predictions at longer distances. To accommodate the data, the model hence favours small values of l. A short length scale, however, leads to oscillations in the GP approximation at the flat parts of the energy curve as the predictive mean between the observation points approaches the mean of the whole data.

In addition to the stationarity of the covariance function, part of the problem is due to the strong smoothness assumptions of the infinitely differentiable squared exponential covariance function. The infinitely differentiable model tends to avoid abrupt changes not only in energy and its first derivatives but also in derivatives of any order. As shown in the Supporting Information of Publication III and also by Denzel and Kästner (2018a), Matérn covariance functions with smoothness parameter $\nu = 3/2$ or $\nu = 5/2$ may perform somewhat better in modelling chemical systems than the squared exponential covariance function but are not able to fully resolve the problem. These covariance functions are here denoted by $k_x^{\mathrm{M}-3/2}$ and $k_x^{\mathrm{M}-5/2}$, respectively:

$$k_x^{\mathrm{M}-3/2}(\mathbf{x},\mathbf{x}') = \sigma_{\mathrm{c}}^2 + \sigma_{\mathrm{m}}^2\big(1+\sqrt{3}\mathscr{D}_x(\mathbf{x},\mathbf{x}')\big)\exp\big(-\sqrt{3}\mathscr{D}_x(\mathbf{x},\mathbf{x}')\big), \tag{2.31}$$

and

$$k_x^{\mathrm{M}-5/2}(\mathbf{x},\mathbf{x}') = \sigma_\mathrm{c}^2 + \sigma_\mathrm{m}^2\Big(1+\sqrt{5}\mathscr{D}_x(\mathbf{x},\mathbf{x}') + \frac{5}{3}\mathscr{D}_x^2(\mathbf{x},\mathbf{x}')\Big)\exp\big(-\sqrt{5}\mathscr{D}_x(\mathbf{x},\mathbf{x}')\big). \quad (2.32)$$

Since potential energy typically changes faster with respect to atom coordinates when the atoms are close to each other, a modified difference measure based on inverse interatomic distances is introduced in Publication III and used also in Publication IV to replace $\mathscr{D}_x(\mathbf{x},\mathbf{x}')$ in the squared exponential covariance function:

$$\mathscr{D}_{1/r}(\mathbf{x},\mathbf{x}') = \sqrt{\sum_{i\in A_\mathrm{m}} \sum_{\substack{j\in A_\mathrm{m}, j>i\\ \vee \\ j\in A_\mathrm{f}}} \frac{\left(\frac{1}{r_{i,j}(\mathbf{x})} - \frac{1}{r_{i,j}(\mathbf{x}')}\right)^2}{l_{\phi(i,j)}^2}}, \quad (2.33)$$

where

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

is the distance between atoms i and j and $l_{\phi(i,j)}$ denotes the length scale for atom pair type $\phi(i,j)$. The outer summation goes through the set of moving atoms A_m, and the inner summation includes all other moving atoms and the possible set of frozen atoms A_f with fixed coordinates. The closer an atom is to another atom, the larger effect a displacement of the atom towards or away from the other atom has on the difference measure. Thus, the difference measure can be interpreted to be stretched when atoms approach each other, which makes the covariance function nonstationary with respect to the atom coordinates and allows faster variation of energy in those directions.

With the modified difference measure $\mathscr{D}_{1/r}(\mathbf{x},\mathbf{x}')$, the squared exponential covariance function gets the following form:

$$k_{1/r}(\mathbf{x},\mathbf{x}') = \sigma_\mathrm{c}^2 + \sigma_\mathrm{m}^2 \exp\left(-\frac{1}{2}\sum_{i\in A_\mathrm{m}} \sum_{\substack{j\in A_\mathrm{m}, j>i\\ \vee \\ j\in A_\mathrm{f}}} \frac{\left(\frac{1}{r_{i,j}(\mathbf{x})} - \frac{1}{r_{i,j}(\mathbf{x}')}\right)^2}{l_{\phi(i,j)}^2}\right). \quad (2.34)$$

Expressions for the partial derivatives of k_x, $k_x^{\mathrm{M}-5/2}$, $k_x^{\mathrm{M}-3/2}$, and $k_{1/r}$ required when dealing with derivative observations and predicting the energy gradient (as described in section 2.4) are given in the Appendix and Supporting Information of Publication III. Figure 2.2 shows a two-dimensional illustration where using the stationary squared exponential covariance function k_x leads to oscillations in spite of a dense grid of observations. The Matérn covariance functions $k_x^{\mathrm{M}-5/2}$ and $k_x^{\mathrm{M}-3/2}$ perform better, but the interpolation is poor especially at the lower left corner of

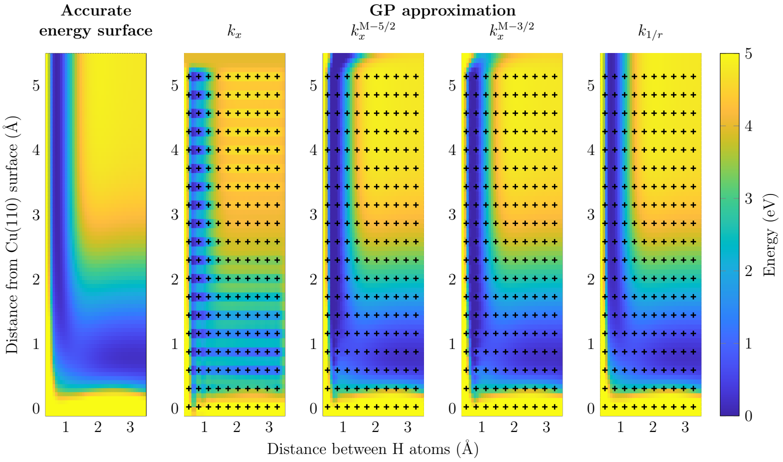


Figure 2.2. A two-dimensional cut through the potential energy surface for a pair of hydrogen atoms near a copper surface (Mills et al., 1995). The GP approximations with optimized hyperparameters are based on accurate values of energy and its first derivatives with respect to coordinates of the hydrogen atoms at the training data points shown with black crosses. With stationary squared exponential (k_x) and Matérn ($k_x^{\mathrm{M}-5/2}$ and $k_x^{\mathrm{M}-3/2}$) covariance functions, the high-gradient observations on the left induce oscillations in the GP approximation. When using covariance function $k_{1/r}$, based on the inverse-distance difference measure $\mathscr{D}_{1/r}$, the training data are interpolated without problems. Figure reproduced with permission from Publication III.

the graph. With the inverse-distance covariance function $k_{1/r}$, the high-gradient observations on the left do not disturb the fitting of the energy surface.

Another advantage of the difference measure $\mathscr{D}_{1/r}$ is that it modifies the similarity structure of the coordinate space in a natural way, which allows more efficient learning with fewer observations. Even more efficient representations for entire potential energy surfaces could be obtained by using some of the carefully designed descriptors or covariance functions associated with the Gaussian approximation potential framework (Bartók et al., 2010; Bartók and Csányi, 2015). These models approximate the total energy of the system with a sum over local atomic environments where the local energy is assumed to be invariant with respect to rotations and translations of the environment and permutations of identical atoms. For example, the SOAP (smooth overlap of atomic positions) covariance function between local environments is based on measuring the overlap in smooth density functions centred at the locations of the neighbouring atoms. In this dissertation, however, the ultimate goal is automated and accurate modelling of the surroudings of a minimum energy path or a saddle point, and the models are therefore kept fairly simple.

3. Methods for finding saddle points

A saddle point of a smooth function is a critical point with a zero gradient but neither local minimum nor maximum point of the function. In this dissertation, the interest is in first-order saddle points of potential energy surfaces located along minimum energy paths, where the Hessian matrix has exactly one negative eigenvalue. In practice, this means that the saddle point is a local maximum point along the direction of the minimum energy path but at a local minimum point along all directions perpendicular to the path.

The two main groups of saddle point search algorithms are chain-of-states methods and surface walking methods. In chain-of-states methods, such as the nudged elastic band method (Mills et al., 1995; Jónsson et al., 1998) or the string method (E et al., 2002; Ren, 2003; E et al., 2007), the task is to find a minimum energy path between the known initial and final states of a transition and to locate the saddle point at the maximum point of that path. The path is represented as a discrete chain of points in the coordinate space, i.e., a chain of states of the system, which is optimized so that the component of the energy gradient perpendicular to the path goes to zero at all points of the chain.

Another group of algorithms, called surface walking methods or mode following methods, aims at finding saddle points without knowing the final state of the transition. The start point for these algorithms is often varied close to a known initial state to search for possible transitions, but it is also common to start closer to the saddle point with an initial guess based for example on approximative minimum energy path calculations. Early examples of this group are based on calculating all eigenvectors of the Hessian matrix and, by modifying the Hessian, maximizing the energy in the direction of the lowest curvature corresponding to the smallest eigenvalue while minimizing the energy in other directions (Cerjan and Miller, 1981; Simons et al. 1983; Banerjee et al., 1985). Some later algorithms, such as the dimer method, find out only the eigenvector corresponding to the smallest eigenvalue without observing the Hessian matrix (Henkelman and Jónsson, 1999; Munro and Wales, 1999; Malek and Mousseau, 2000) and proceed towards the saddle point based on a modified gradient. This approach is often referred to as the minimum mode following method.

Recent advances of saddle point search methods are reviewed in a book chapter by Ásgeirsson and Jónsson (2018). In this dissertation, the focus is on the nudged elastic band method and the dimer method, which are common representatives of the two main groups, both based on the first derivatives of the energy surface. In publications I–IV, these methods are used as parts of the GP-NEB and GP-dimer algorithms, where the search of a minimum energy path and/or a saddle point is enhanced using Gaussian process regression. Similar algorithms can, however, be applied to accelerate practically any other stable saddle point search method.

3.1 Nudged elastic band method

Consider a system of N_m moving atoms with configurations represented by $3N_\mathrm{m}$-dimensional vectors including the atom coordinates. In the nudged elastic band method (Mills et al., 1995; Jónsson et al., 1998), two given local minimum points representing the initial and final states of a transition are connected with a discrete chain of N_im configurations, often referred to as images of the system. The first image of the chain, $\mathbf{R}_0$, is fixed to the initial state and the last image, $\mathbf{R}_{N_\mathrm{im}-1}$, to the final state, whereas the intermediate images, $\mathbf{R}_i, i = 1,2,\ldots,N_\mathrm{im}-2$, are iteratively moved towards a minimum energy path. The simplest path to begin with is obtained by placing the intermediate images regularly along a straight line between $\mathbf{R}_0$ and $\mathbf{R}_{N_\mathrm{im}-1}$. In some cases, however, this may lead to unphysical configurations with overlapping atoms. A better initial guess that avoids the overlapping can be obtained with the IDPP (image dependent pair potential) method (Smidstrup et al., 2014), which aims to place the intermediate images so that the distances between neighbouring atoms change as linearly as possible along the chain, or the geodesic approach recently introduced by Zhu et al. (2019).

The movements of the intermediate images $\mathbf{R}_i, i = 1,2,\ldots,N_\mathrm{im}-2$, are based on the energy $E(\mathbf{R}_i)$, the atomic force vector $\mathbf{F}(\mathbf{R}_i) = -\nabla E(\mathbf{R}_i)$ given by the negative gradient of the energy, and the tangent of the path, $\hat{\boldsymbol{\tau}}_i$. The goal of the movements is to zero an effective force vector, here referred to as the NEB force:

$$\mathbf{F}_i^\mathrm{NEB} = \mathbf{F}^\perp(\mathbf{R}_i) + \mathbf{F}_i^\mathrm{s}, \tag{3.1}$$

where

$$\mathbf{F}^\perp(\mathbf{R}_i) = \mathbf{F}(\mathbf{R}_i) - (\mathbf{F}(\mathbf{R}_i)\cdot\hat{\boldsymbol{\tau}}_i)\hat{\boldsymbol{\tau}}_i \tag{3.2}$$

is the component of $\mathbf{F}(\mathbf{R}_i)$ perpendicular to the normalized path tangent $\hat{\boldsymbol{\tau}}_i$ at $\mathbf{R}_i$ and $\mathbf{F}_i^\mathrm{s}$ is a spring force parallel to $\hat{\boldsymbol{\tau}}_i$. In the original formulation (Jónsson et al., 1998), the spring force is defined as

$$\mathbf{F}_i^{\mathrm{s}} = \left(\left(k_{i+1}^{\mathrm{s}}(\mathbf{R}_{i+1} - \mathbf{R}_i) - k_i^{\mathrm{s}}(\mathbf{R}_i - \mathbf{R}_{i-1})\right) \cdot \hat{\boldsymbol{\tau}}_i\right) \hat{\boldsymbol{\tau}}_i, \tag{3.3}$$

where k_i^{s} is a spring constant that determines the relative length desired for the interval between images $\mathbf{R}_i$ and $\mathbf{R}_{i-1}$. A common choice of equal intervals is made for applications of NEB in Publication I, where the spring force is defined according to Henkelman and Jónsson (2000) as

$$\mathbf{F}_i^{\mathrm{s}} = k^{\mathrm{s}}(||\mathbf{R}_{i+1} - \mathbf{R}_i|| - ||\mathbf{R}_i - \mathbf{R}_{i-1}||)\hat{\boldsymbol{\tau}}_i. \tag{3.4}$$

The word *nudged* refers to the separation of the forces into two orthogonal components, which is an essential feature of the NEB method. Removal of the atomic force component parallel to the path prevents the images from sliding down towards the minimum energy points and leaves the control of the distribution of the images along the path to the spring forces. On the other hand, projection of the spring force on the path tangent prevents corner cutting since the perpendicular spring forces would tend to straighten the path at curves. A small perpendicular spring force can sometimes stabilize the path optimization by preventing kinks of the path in regions where the atomic forces perpendicular to the path are small compared to the forces along the path, but these solutions require some sort of switching function for the magnitude of the force (Jónsson et al., 1998; Trygubenko and Wales, 2004; Sheppard et al., 2008; Maras et al., 2016). Another cure for this behaviour is obtained by modifying the estimate of the path tangent (Henkelman and Jónsson, 2000). Whereas a simple estimate for the path tangent is parallel to a line segment connecting the previous and the following image,

$$\hat{\boldsymbol{\tau}}_i = \frac{\mathbf{R}_{i+1} - \mathbf{R}_{i-1}}{||\mathbf{R}_{i+1} - \mathbf{R}_{i-1}||}, \tag{3.5}$$

a better-behaved estimate for the direction of the tangent, used also in Publications I–III, can be achieved by

$$\boldsymbol{\tau}_i = \begin{cases} \mathbf{R}_{i+1} - \mathbf{R}_i, & \text{if } E(\mathbf{R}_{i-1}) < E(\mathbf{R}_i) < E(\mathbf{R}_{i+1}) \\ \mathbf{R}_i - \mathbf{R}_{i-1}, & \text{if } E(\mathbf{R}_{i+1}) < E(\mathbf{R}_i) < E(\mathbf{R}_{i-1}) \\ \Delta E_-(\mathbf{R}_{i+1} - \mathbf{R}_i) + \Delta E_+(\mathbf{R}_i - \mathbf{R}_{i-1}), & \text{if } E(\mathbf{R}_{i\pm1}) < E(\mathbf{R}_i) \\ \Delta E_+(\mathbf{R}_{i+1} - \mathbf{R}_i) + \Delta E_-(\mathbf{R}_i - \mathbf{R}_{i-1}), & \text{if } E(\mathbf{R}_i) < E(\mathbf{R}_{i\pm1}), \end{cases} \tag{3.6}$$

where $\Delta E_- = |E(\mathbf{R}_i) - E(\mathbf{R}_{i-1})|$ and $\Delta E_+ = |E(\mathbf{R}_{i+1}) - E(\mathbf{R}_i)|$. If the energy at an image is either higher or lower than at both of its neighbours, the direction of the tangent is defined as a weighted average of two line segments. Otherwise, only the line segment to the neighbouring image with higher energy is taken into account.

The ultimate goal of NEB calculations is often to locate the saddle point at the maximum point of the minimum energy path. However, the maximum energy may be under- or overestimated if interpolated based on the discrete representation of the path. The climbing image nudged elastic band (CI-NEB) method (Henkelman, Uberuaga, and Jónsson, 2000) provides a solution to this problem by letting one of the images climb upwards along the path towards the saddle point. The method is often started with regular NEB iterations to find a rough shape of the path, and the image with the highest energy is then selected as the climbing image $\mathbf{R}_{i_{\mathrm{CI}}}$. This special image is not exposed to any spring forces, but a component of the atomic force perpendicular to the path tangent is restored and reverted to point towards the direction of increasing energy along the path. The effective force on the climbing image is hence given as

$$\mathbf{F}_{i_{\mathrm{CI}}}^{\mathrm{NEB}} = \mathbf{F}(\mathbf{R}_{i_{\mathrm{CI}}}) - 2(\mathbf{F}(\mathbf{R}_{i_{\mathrm{CI}}}) \cdot \hat{\boldsymbol{\tau}}_{i_{\mathrm{CI}}})\hat{\boldsymbol{\tau}}_{i_{\mathrm{CI}}}. \tag{3.7}$$

With equal spring constants, the images leaving on each side of the climbing image are then distributed evenly on each subpath. Since the saddle point is usually the most interesting part of the minimum energy path, it is common to set a tighter convergence threshold for the NEB force of the climbing image than for the rest of the images. Figure 3.1 illustrates the effect of the climbing image on a NEB calculation on an artificial two-dimensional energy surface (Müller and Brown, 1979). Without the climbing image feature, the images of the converged path are evenly distributed and miss the saddle point found by the climbing image. The CI-NEB method has a central role in Publication II, where the details of the GP-NEB algorithm are modified to take the climbing image into account.

The simplest way to define the NEB iterations is to move the images in the direction of the NEB force with a step length proportional to the magnitude of the NEB force. This steepest descent approach may, however, require an excessively large number of iterations. A more efficient control of the step length is obtained by the velocity projection optimization

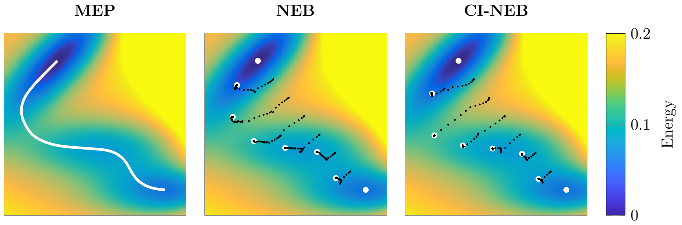


Figure 3.1. Progression of a NEB calculation on a two-dimensional Müller-Brown energy surface (Müller and Brown, 1979) with and without the climbing image feature. The white dots represent images of the converged path, and the small black dots represent earlier locations of intermediate images where energy and its first derivatives have been evaluated during the process. With CI-NEB, the third image of the path converges to the saddle point, whereas the evenly distributed NEB path takes a shortcut on the critical area. The continuous minimum energy path is presented on the left.

(VPO) algorithm (also known as quick-min) based on molecular dynamics (Jónsson et al., 1998). The movement of the images is accelerated based on the velocity Verlet algorithm (Andersen, 1980; Swope et al., 1982), or alternatively a simpler Euler integrator (Sheppard et al., 2008), with the exception that the velocity vector is projected on the direction of the NEB force or zeroed if the direction of the projected velocity would be opposite to the NEB force. This optimization method is used in the implementation of the GP-NEB algorithm in Publications I–III.

The lack of a well-defined objective function due to the force projections make NEB challenging for more advanced optimization methods, such as nonlinear conjugate gradient (Fletcher and Reeves, 1964; Polak and Ribière, 1969) or limited memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) algorithms (Nocedal, 1980; Liu and Nocedal, 1989), that modify also the search direction based on previous iterations and often perform a line search along that direction based on a finite difference step. These methods have a potential to faster convergence to the minimum energy path especially when the convergence threshold is tight but may be unstable during the early phases of the optimization (Sheppard et al., 2008). Better convergence properties can be achieved also by using the fast inertial relaxation engine (Bitzek et al., 2006), shortened as FIRE, which is an extension of the VPO algorithm involving adaptive time steps and additional modifications to the velocity. Since acquisition of the second derivatives of energy is usually too expensive, the optimization algorithm is required to be based only on the first derivatives. In case accurate observations of the second derivatives were easily available, the NEB optimization could be done efficiently with a Newton-Rapshon method using analytical calculations of the derivatives of the NEB forces (Bohner et al., 2013).

3.2 Dimer method

The dimer method (Henkelman and Jónsson, 1999) is an example of a minimum mode following algorithm with the objective to find a saddle point by following the direction of the lowest curvature on the energy surface without knowing the final state of the transition. Inspired by the idea of Voter (1997), the lowest curvature mode is found by rotating a dimer consisting of a pair of images, $\mathbf{R}_1$ and $\mathbf{R}_2$, with respect to its middle point $\mathbf{R}_0$, and the whole dimer is then translated towards the saddle point based on a force vector where the component parallel to the direction of the dimer is reverted to point towards the direction of increasing energy, similarly as in the CI-NEB method. During the algorithm, rotation and translation phases alternate until the magnitude of the translational force is below some convergence threshold. Figure 3.2 shows a simple two-dimensional

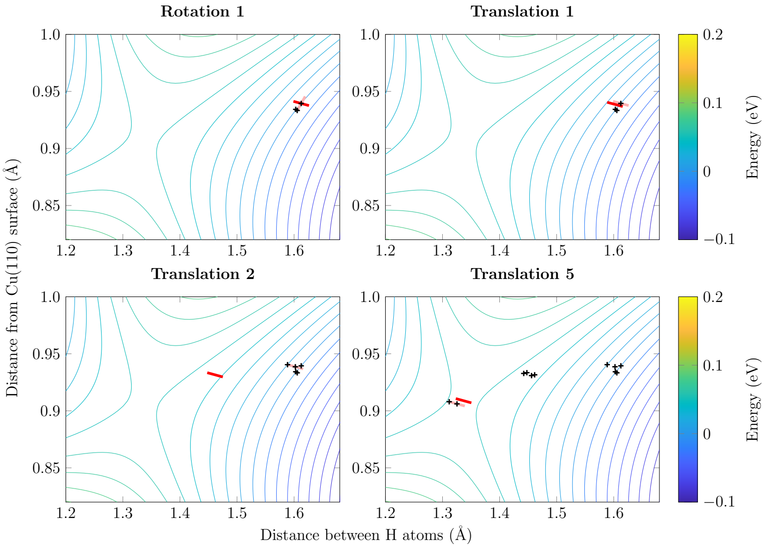


Figure 3.2. Progression of the dimer method in a simple example where two hydrogen atoms are free to move near a fixed copper surface (Mills et al., 1995). The saddle point and the initial dimer coincide with the same two-dimensional cut of the coordinate space as shown in figure 2.2. The pink and red bars represent the dimer before and after the rotation or translation, respectively, and the black crosses represent locations where energy and its first derivatives have been evaluated during the process. In this case, the orientation of the dimer after the first rotation turns out to be close enough to the lowest curvature mode of the saddle point so that no further rotations are needed.

illustration of the progression of the dimer method in the same system as shown in figure 2.2.

The rotations towards the lowest curvature mode are based on the atomic force vectors $\mathbf{F}(\mathbf{R}_1) = -\nabla E(\mathbf{R}_1)$ and $\mathbf{F}(\mathbf{R}_2) = -\nabla E(\mathbf{R}_2)$, given by the negative energy gradient at the two images. The distance from the middle point $\mathbf{R}_0$ to $\mathbf{R}_1$ and $\mathbf{R}_2$, referred to as the dimer separation $\Delta_{\mathbf{R}}$, is fixed to a small value in order to estimate the second derivative of energy along the dimer as accurately as possible. The direction of the lowest curvature corresponds to the orientation where the dimer energy, defined as $E(\mathbf{R}_1)+E(\mathbf{R}_2)$, is minimized. The minimum curvature mode is thus found by zeroing a scaled rotational force given as

$$\mathbf{F}_{\text{rot}} = \frac{\mathbf{F}^{\perp}(\mathbf{R}_1) - \mathbf{F}^{\perp}(\mathbf{R}_2)}{\Delta_{\mathbf{R}}}, \tag{3.8}$$

where

$$\mathbf{F}^{\perp}(\mathbf{R}_i) = \mathbf{F}(\mathbf{R}_i) - (\mathbf{F}(\mathbf{R}_i)\cdot\hat{\mathbf{N}})\hat{\mathbf{N}} \tag{3.9}$$

is the component of $\mathbf{F}(\mathbf{R}_i)$ perpendicular to the orientation vector $\hat{\mathbf{N}}$, which is a unit vector pointing from $\mathbf{R}_0$ towards $\mathbf{R}_1$. Instead of evaluating the force at both $\mathbf{R}_1$ and $\mathbf{R}_2$ between subsequent rotations, it is more efficient to evaluate the force at the fixed middle point $\mathbf{R}_0$ and extrapolate the force at $\mathbf{R}_2$ as $\mathbf{F}(\mathbf{R}_2) \approx 2\mathbf{F}(\mathbf{R}_0) - \mathbf{F}(\mathbf{R}_1)$, as suggested by Olsen et al. (2004).

The rotational plane for each rotation iteration is spanned by the orientation vector $\hat{\mathbf{N}}$ and another unit vector $\hat{\boldsymbol{\Omega}}$, which defines the direction of the rotation for $\mathbf{R}_1$. The steepest descent direction is simply the direction of the rotational force: $\hat{\boldsymbol{\Omega}} = \mathbf{F}_{\text{rot}}/||\mathbf{F}_{\text{rot}}||$. It is also possible to use a nonlinear conjugate gradient (Fletcher and Reeves, 1964; Polak and Ribière, 1969; Henkelman and Jónsson, 1999) or a more efficient L-BFGS (Nocedal, 1980; Liu and Nocedal, 1989; Kästner and Sherwood, 2008) approach, where $\hat{\boldsymbol{\Omega}}$ is modified based on previous rotation iterations.

In the original formulation (Henkelman and Jónsson, 1999), a small preliminary step with a rotation angle ω^* is first taken to get a finite difference approximation to the derivative of the rotational force, and the optimal rotation angle ω is then obtained based on a local quadratic approximation to the energy surface. Heyden et al. (2005) prefer a larger preliminary rotation instead of a finite difference step in order to avoid possible problems with noisy data. They suggest the following rough estimate, used also in Publication IV, for the preliminary rotation angle:

$$\omega^* = \frac{1}{2}\arctan\frac{(\mathbf{F}(\mathbf{R}_1) - \mathbf{F}(\mathbf{R}_0)) \cdot \hat{\boldsymbol{\Omega}}}{\Delta_{\mathbf{R}}|C|}, \tag{3.10}$$

where

$$C = (\mathbf{F}(\mathbf{R}_0) - \mathbf{F}(\mathbf{R}_1)) \cdot \hat{\mathbf{N}}/\Delta_{\mathbf{R}} \tag{3.11}$$

is an estimate for the curvature of energy along the dimer. The dimer orientation and rotation direction after the preliminary rotation step are given by

$$\hat{\mathbf{N}}^* = \hat{\mathbf{N}}\cos\omega^* + \hat{\boldsymbol{\Omega}}\sin\omega^* \tag{3.12}$$

and

$$\hat{\boldsymbol{\Omega}}^* = -\hat{\mathbf{N}}\sin\omega^* + \hat{\boldsymbol{\Omega}}\cos\omega^*, \tag{3.13}$$

and the force $\mathbf{F}(\mathbf{R}_1^*)$ is then evaluated at $\mathbf{R}_1^* = \mathbf{R}_0 + \Delta_{\mathbf{R}}\hat{\mathbf{N}}^*$. Based on a local quadratic approximation, the final rotation angle that minimizes the dimer energy on the rotational plane is given as

$$\omega = \begin{cases} \frac{1}{2}\arctan\frac{b_1}{a_1}, & \text{if } \frac{b_1}{a_1} \geq 0 \\ \frac{1}{2}\arctan\frac{b_1}{a_1} + \frac{\pi}{2}, & \text{if } \frac{b_1}{a_1} < 0, \end{cases} \tag{3.14}$$

where

$$b_1 = (\mathbf{F}(\mathbf{R}_0) - \mathbf{F}(\mathbf{R}_1)) \cdot \hat{\boldsymbol{\Omega}}/\Delta_{\mathbf{R}} \tag{3.15}$$

and

$$a_1 = \frac{b_1\cos(2\omega^*) - (\mathbf{F}(\mathbf{R}_0) - \mathbf{F}(\mathbf{R}_1^*)) \cdot \hat{\boldsymbol{\Omega}}^*/\Delta_{\mathbf{R}}}{\sin(2\omega^*)}. \tag{3.16}$$

The new dimer orientation after the rotation is given by

$$\hat{\mathbf{N}}^{\text{new}} = \hat{\mathbf{N}} \cos\omega + \hat{\boldsymbol{\Omega}} \sin\omega \tag{3.17}$$

and the new $\mathbf{R}_1$ by

$$\mathbf{R}_1^{\text{new}} = \mathbf{R}_0 + \Delta_{\mathbf{R}} \hat{\mathbf{N}}^{\text{new}}. \tag{3.18}$$

In some implementations, no more than one rotation iteration is performed between the translation steps. The rotation iterations can be also repeated until rotational convergence, defined based on the preliminary or final rotation angle or the magnitude of rotational force, or until some maximum number of consecutive rotations is reached. In that case, the number of force evaluations between consecutive rotation iterations can be reduced by extrapolating $\mathbf{F}(\mathbf{R}_1^{\text{new}})$ from $\mathbf{F}(\mathbf{R}_0)$, $\mathbf{F}(\mathbf{R}_1)$, and $\mathbf{F}(\mathbf{R}_1^*)$ and using this estimate when calculating the rotational force for the following rotation iteration (Kästner and Sherwood, 2008).

After each rotation phase, a translation step is performed to move the middle point of the dimer towards the saddle point. The nature of the translation step depends on the curvature along the current orientation vector $\hat{\mathbf{N}}$, estimated either by the quadratic approximation (Olsen et al., 2004; Heyden et al., 2005) or equation 3.11. If the curvature is positive, the dimer is assumed to be in a convex region with positive second derivatives of energy in all directions, which is often the case if the start point is chosen to be close to an minimum energy point. In this case, a step with some predifined length is taken to the direction of increasing energy along $\hat{\mathbf{N}}$ to make the dimer climb up from the convex region. If the curvature along the dimer is negative, the translational force is obtained by reverting the component of $\mathbf{F}(\mathbf{R}_0)$ parallel to the dimer:

$$\mathbf{F}_{\text{trans}} = \mathbf{F}(\mathbf{R}_0) - 2\mathbf{F}^{\parallel}(\mathbf{R}_0), \tag{3.19}$$

where

$$\mathbf{F}^{\parallel}(\mathbf{R}_0) = (\mathbf{F}(\mathbf{R}_0) \cdot \hat{\mathbf{N}})\hat{\mathbf{N}}. \tag{3.20}$$

This allows the dimer to climb upwards on the energy surface following the direction of the minimum curvature mode mode while minimizing the energy in directions perpendicular to the dimer. The displacement of $\mathbf{R}_0$ can be performed using any gradient-based optimization approach, including nonlinear conjugate gradient (Fletcher and Reeves, 1964; Polak and Ribière, 1969) and L-BFGS (Nocedal, 1980; Liu and Nocedal, 1989) algorithms. Similarly as in the rotation phase, a preliminary step can be taken to estimate a proper step length for the translation. In the L-BFGS approach, however, a good estimate for the translation step length is provided by an inverse Hessian approximated implicitly based on information stored during previous translation iterations. As suggested by Kästner and Sherwood (2008), the L-BFGS approach is applied to both translations and rotations in the GP-dimer algorithm presented in Publication IV.

4. Summary of contributions

The contribution of this dissertation consists of development and testing of two algorithms that utilize Gaussian process regression to enhance searches of saddle points and minimum energy paths. The GP-NEB algorithm aims to find a minimum energy path between two known minimum energy configurations and the saddle point located at the maximum point of the path, whereas the GP-dimer algorithm only searches for the saddle point starting somewhere from its vicinity.

4.1 GP-NEB algorithm

The general idea of using machine learning methods to enhance saddle point search algorithms has been introduced by Peterson (2016), who applied artificial neural networks to nudged elastic band calculations. In the iterative procedure, a minimum energy path is first found on an approximate energy surface based on a machine learning model, and accurate evaluations of energy and its first derivatives are then performed to check if the path has converged also on the accurate energy surface. If the convergence criteria are not satisfied, the new observations are included in the training data set, the machine learning model is updated, and the path is relaxed again on the approximate energy surface. The iterations are repeated until final convergence is confirmed by accurate evaluations. The advantage of this approach is based on the assumption that the accurate evaluations are significantly more expensive than training of the machine learning model or approximation of energy and its derivatives based on the model. By performing the path relaxation on the approximate energy surface, the total number of accurate evaluations required for convergence can be reduced and the minimum energy path search hence accelerated.

Publication I presents an initial step in the development of a similar algorithm where Gaussian process regression is applied instead of neural networks as a machine learning approach to model the energy surface. Whereas optimization of a large number of weights of a neural network model may be challenging due to many local minima of the cost function, optimization of the hyperparameters of a Gaussian process model is a much easier task. As described in section 2.4, Gaussian process models allow straightforward ways to handle derivatives, which is beneficial when learning from the derivative observations and predicting NEB forces for the path relaxation. Furthermore, GP models have been shown to perform well especially when learning from small training data sets, which makes them appealing for this application. In Publication I, the feasibility of the GP-NEB approach is demonstrated for three simple benchmark transitions, where two atoms of an heptamer island move to adjacent sites on the (111) surface of a FCC (face-centred cubic) crystal (Henkelman, Jóhannesson, and Jónsson, 2000; Chill et al., 2014). As compared with a regular NEB method, the number of evaluations required for convergence to the minimum energy path is decreased to less than fifth with a simple implementation of the GP-NEB algorithm using the stationary squared exponential covariance function k_x (see equation 2.29 in section 2.5).

Publication II extends the GP-NEB method to CI-NEB calculations and presents detailed descriptions for two variants of the algorithm. The simpler one, referred to as the all-images-evaluated (AIE) algorithm, follows the original idea of Peterson (2016) by evaluating accurate energy and its first derivatives at all intermediate images of the NEB path relaxed on the approximate energy surface. Figure 4.1 shows the progression of the AIE algorithm in the same two-dimensional task as shown in figure 3.1 for the regular CI-NEB method. Knowing the coordinates of the initial path, $\mathbf{R}_i, i = 0,1,\ldots,N_{\mathrm{im}}-1$, and accurate energy and its (zero) gradient at the two end points, $E(\mathbf{R}_0)$, $\nabla E(\mathbf{R}_0)$, $E(\mathbf{R}_{N_{\mathrm{im}}-1})$, and $\nabla E(\mathbf{R}_{N_{\mathrm{im}}-1})$, the algorithm is started by evaluating $E(\mathbf{R}_i)$ and $\nabla E(\mathbf{R}_i)$ at the intermediate images $\mathbf{R}_i, i = 1,\ldots,N_{\mathrm{im}}-2$. Based on these data, a GP model for the energy surface is trained by optimizing the hyperparameters of the covariance function, and a CI-NEB calculation is performed using the mean of the

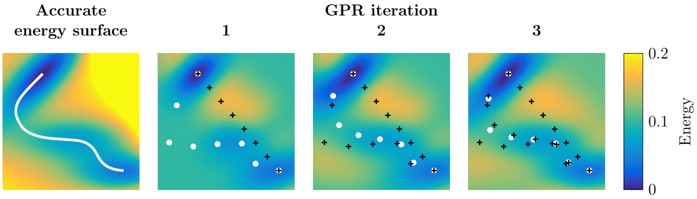


Figure 4.1. Progression of the simpler all-images-evaluated (AIE) version of the GP-NEB algorithm on a two-dimensional Müller-Brown energy surface (Müller and Brown, 1979). The white dots represent images of the relaxed CI-NEB path on an approximate energy surface obtained by GP regression. After each GPR iteration, final convergence of the path is checked by evaluating accurate energy and its first derivatives at all intermediate images, and those observations are then added to the training data set (observed locations marked with black crosses). Figure reproduced with permission from Publication II.

posterior predictive distribution of energy and its derivatives (see equations 2.25 and 2.27 in section 2.4) when calculating the NEB forces. After the CI-NEB path has relaxed on the approximate energy surface, new energy and gradient evaluations are made at the intermediate images of the relaxed path and added to the training data set for the following GPR iteration. The algorithm is continued until final convergence criteria for the accurate NEB forces are satisfied after three GPR iterations and a total of 24 evaluations.

Due to the probabilistic nature of Gaussian process regression, the predictions of energy and its derivatives are expressed as probability distributions. The more advanced variant of GP-NEB, referred to as the one-image-evaluated (OIE) algorithm, utilizes the variance of the posterior distribution of energy (see equation 2.26 in section 2.4) as a measure of uncertainty to direct the evaluations to locations where they are most useful. According to the main rule, accurate energy and derivatives are evaluated only at the image with the highest uncertainty before updating the GP model and relaxing the path. However, since confirmation of the final convergence requires accurate energy gradient to be known for all images of the path, also the other intermediate images are included in the evaluations one by one without moving the path as long as there is a chance that final convergence might have been reached based on the mixture of accurate and approximated NEB forces. Since the convergence criterion may be tighter for the climbing image and since its location affects

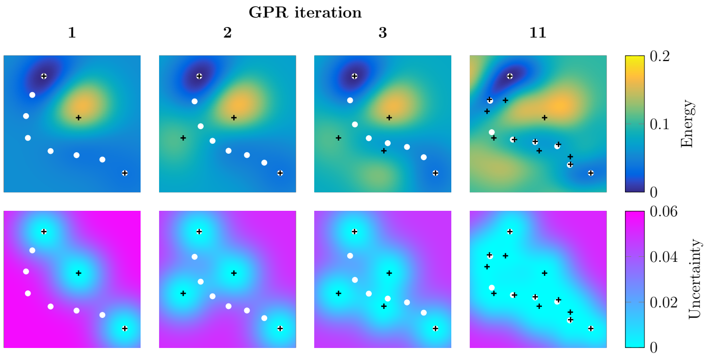


Figure 4.2. Progression of the more advanced one-image-evaluated (OIE) version of the GP-NEB algorithm on a two-dimensional Müller-Brown energy surface (Müller and Brown, 1979). The white dots represent images of the relaxed CI-NEB path on an approximate energy surface obtained by GP regression. The lower panel shows the standard deviation of the posterior distribution of energy representing the uncertainty of the predictions according to the GP model. After GPR iterations 1, 2, and 3, accurate energy and its first derivatives are evaluated at the image with the largest uncertainty, and the information is then added to the training data set (observed locations marked with black crosses). After GPR iteration 11, the path is not moved anymore but the final convergence is confirmed by accurate evaluations at each of the intermediate images. Figure reproduced with permission from Publication II.

on the distribution of the other images, the climbing image is favoured over other images in the evaluation order of the convergence check. Figure 4.2 shows the progression of the OIE algorithm in the two-dimensional example task. The final shape of the path is here obtained after eleven energy and gradient evaluations, and the final convergence is then confirmed by six more evaluations.

In Publication II, the two variants of the GP-NEB algorithm are tested in CI-NEB calculations for the whole heptamer island benchmark (Henkelman, Jóhannesson, and Jónsson, 2000; Chill et al., 2014) including thirteen transitions. As compared with a regular CI-NEB method, the number of evaluations required for convergence to the minimum energy path is decreased by an order of magnitude. The OIE algorithm reduces the number of evaluations to about a half of what is required by the AIE algorithm. As an additional test feature, information about the second derivatives of energy at the two end points are included by adding finite difference data points in the initial training data set. Since the Hessian of energy is often evaluated anyway at the initial and final states of the transition when calculating transition rates using the harmonic approximation to the transition state theory (Vineyard, 1957), these evaluations may be considered available without additional effort. The use of the Hessian data reduces the number of observations by about 20% when using the AIE algorithm, but the effect is smaller for the OIE algorithm.

In Publications I and II, a simple GP model with a stationary squared exponential covariance function k_x is successfully used in the GP-NEB calculations. In some systems, however, the stationarity of the covariance function with respect to the atom coordinates may lead to problems as illustrated in section 2.5. In Publication III, these problems are avoided by defining a modified covariance function $k_{1/r}$ where the difference measure fed to the squared exponential covariance function is based on differences in the inverse interatomic distances (see equation 2.34 in section 2.5). This difference measure stretches when atoms are closer to each other, which makes it easier to model large repulsive forces. In addition, the more informative covariance structure allows more efficient learning of the potential energy surface.

Even though the modified GP model handles well also large repulsive forces, avoiding unphysical configurations and constraining the exploration of uncertain regions may still stabilize the algorithm. Another modification introduced in Publication III concerns the early stopping criteria that define the allowed region for the images of the path during the NEB relaxation phase. The early stopping criterion used in Publication II is based on the distance to the nearest observed data point according to the regular difference measure $\mathscr{D}_x$ (see equation 2.30 in section 2.5) with the limit set to a half of the length of the initial path. If the limit is exceeded, then the last step of the NEB relaxation phase is rejected, the relaxation phase is stopped, and the following evaluation is performed at the image that violated the condition. In Publication III, an additional early stopping criterion is introduced based on relative changes in the interatomic distances. The condition requires that for each image of the current path, there exists an observed data point with all interatomic distances between 2/3 and 3/2 of the corresponding distance in the current image. Accompanied with the inverse-distance covariance function $k_{1/r}$,

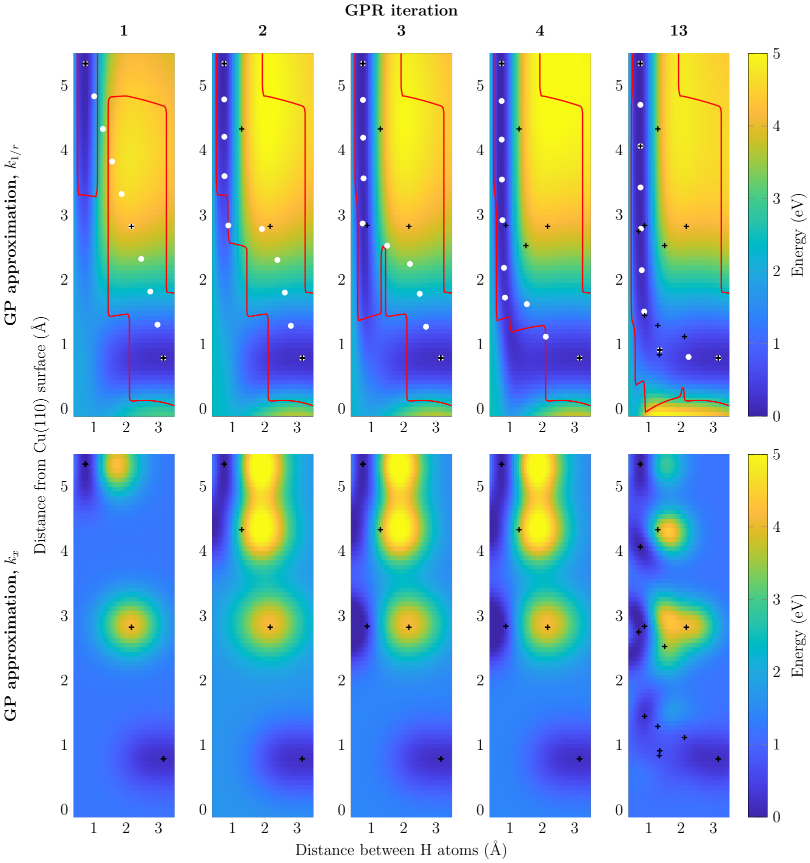


Figure 4.3. A two-dimensional cut through the potential energy surface for a pair of hydrogen atoms near a fixed copper surface (Mills et al., 1995). The upper panel shows the progression of the modified GP-NEB algorithm. The white dots are projections of the images of the relaxed CI-NEB path on an approximate energy surface obtained by GP regression with the inverse-distance covariance function $k_{1/r}$, and the red line shows the border of the allowed region defined by the accompanying early stopping criterion. The black crosses are projections of the training data points. In the first four GPR iterations, the NEB relaxation phase is terminated by the early stopping rule, and the final path is obtained after thirteen GPR iterations. For comparison, the lower panel shows GP approximations with the stationary covariance function k_x using the same training data sets as in the upper panel. Figure reproduced with permission from Publication III.

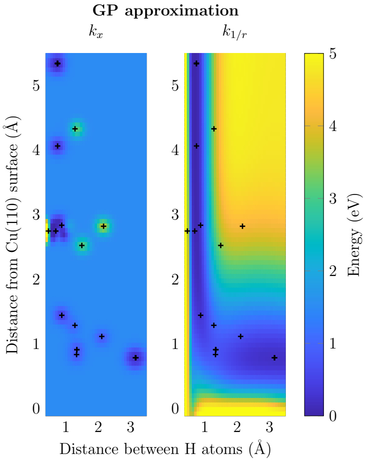


Figure 4.4. GP approximations based on covariance functions k_x and $k_{1/r}$ corresponding to the rightmost graphs in figure 4.3 with a high-gradient data point close to the left border of the graph added to the training data set. Figure reproduced with permission from Publication III.

this criterion practically prevents the distance between two atoms from becoming shorter than 2/3 of the unknown bond length (notice that this does not apply with a stationary covariance function).

The upper panel of figure 4.3 shows the progression of the modified GP-NEB algorithm in a CI-NEB calculation for a dissociation of a hydrogen molecule on a fixed copper surface (Mills et al., 1995). The initial and final states coincide with the same two-dimensional cut of the six-dimensional coordinate space as shown in figure 2.2. The GP approximation based on the inverse-distance covariance function looks quite realistic already in the beginning, when the training data include the energy and its first derivatives at one intermediate image and the two end points in addition to the Hessian data at the end points. Since the third image of the initial path is outside the allowed regions, the early stopping rule is triggered already before moving the images, and also the NEB relaxations in the following three GPR iterations are terminated by the early stopping rule. The final convergence is confirmed after nineteen energy and gradient evaluations, whereas a regular CI-NEB calculation requires about 500 evaluations.

For comparison, the lower panel of figure 4.3 shows what the GP approximation with the same training data would look like if the stationary squared exponential covariance function k_x was used instead of the inverse-distance covariance function $k_{1/r}$. Since the stationary GP model extrapolates the attractive forces acting on the hydrogen atoms to regions where the atoms collide, it would be difficult to keep the images away from regions of large repulsive forces without a too restrictive stopping rule. As shown in figure 4.4, an additional data point from the repulsive region would make interpolation of the training data set more difficult for the stationary model and lead to a short length scale. With the inverse-distance covariance function, the additional data point would not cause problems.

In addition to demonstrations in systems that are challenging for stationary GP models, Publication III reports results also for the heptamer island benchmark for which the squared exponential covariance function k_x works well. Figure 4.5 shows the average number of energy and gradient evaluations required for GP-NEB calculations with a varying number of degrees of freedom. Depending on the algorithm variant, the inverse-distance covariance function with the accompanying early stopping criterion reduces the number of energy and force evaluations by about 30–50% when compared with the squared exponential covariance function.

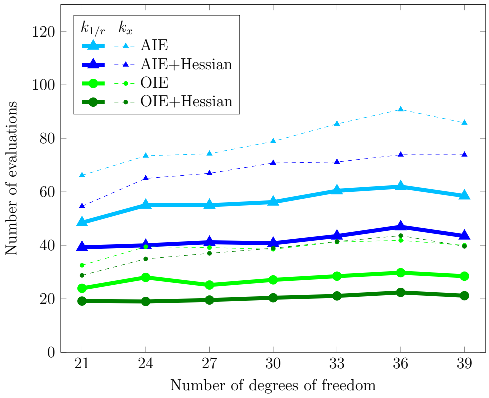


Figure 4.5. Number of energy and gradient evaluations required for convergence of CI-NEB calculations in a heptamer island benchmark (Henkelman, Jóhannesson, and Jónsson, 2000; Chill et al., 2014) with variants of the GP-NEB algorithm. The average over thirteen different transitions is presented as a function of the number of degrees of freedom, increased by allowing a larger number of substrate atoms to move. The narrow dashed lines present results with the stationary squared exponential covariance function k_x, and the thick solid lines present the corresponding results when using the inverse-distance covariance function $k_{1/r}$ with the accompanying stopping criterion. The blue triangles represent the all-images-evaluated (AIE) algorithm, and the green dots represent the one-image-evaluated (OIE) algorithm. The use of Hessian data at the two end points is indicated by darker colour. Figure reproduced with permission from Publication III.

4.2 GP-dimer algorithm

Publication IV applies the Gaussian process regression approach used in the GP-NEB algorithm to the dimer method in a saddle point search task where only a start point is known. A similar general scheme connecting Gaussian process regression with surface walking methods has been recently applied by Denzel and Kästner (2018b), who use a stationary Matérn-5/2 covariance function to build a computationally efficient multilevel Gaussian process model (Denzel and Kästner, 2018a). The GP-dimer algorithm in Publication IV applies the more expressive inverse-distance covariance function $k_{1/r}$ coupled with the robust stopping criterion as suggested for the GP-NEB algorithm in Publication III, and the performance is compared with corresponding results obtained with stationary covariance functions.

With the middle point $\mathbf{R}_0$ of the initial dimer set to the given start point and images $\mathbf{R}_1$ and $\mathbf{R}_2$ aligned with the a possibly randomized start orientation, the GP-dimer algorithm is started by evaluating accurate energy and its gradient at $\mathbf{R}_0$ and $\mathbf{R}_1$, i.e., $E(\mathbf{R}_0)$, $\nabla E(\mathbf{R}_0)$, $E(\mathbf{R}_1)$, and $\nabla E(\mathbf{R}_1)$. If no information is available about the energy surface or the direction of the lowest energy curvature at the start point, it is useful to perfom initial rotations with accurate evaluations before translating the dimer. In the GP-dimer algorithm, this initial phase is performed by repeated initial rotation rounds on an approximate energy surface obtained by GP regression based on the evaluations made so far. During each initial rotation round, the direction of the lowest curvature on the approximate energy surface is found according to a regular rotation scheme using the mean of the posterior predictive distribution of the energy gradient to calculate the rotational force (see equation 2.27 in section 2.4), and accurate energy $E(\mathbf{R}_1)$ and gradient $\nabla E(\mathbf{R}_1)$ are then evaluated at the new location of $\mathbf{R}_1$. The initial rotation phase is stopped when the preliminary rotation angle ω^* (see equation 3.10 in section 3.2) based on the accurate gradients $\nabla E(\mathbf{R}_0)$ and $\nabla E(\mathbf{R}_1)$ or the angle between the relaxed orientations of two subsequent rounds is below a given threshold. As shown in Publication IV, this approach requires fewer evaluations for rotational convergence than regular rotation schemes. A similar initial phase where GP regression is utilized to find the direction of the lowest energy curvature is applied also by Denzel and Kästner (2018b).

In the actual GPR iterations started after the initial rotation phase, the dimer is both rotated and translated based on the GP approximation. During each GPR iteration, a saddle point on the approximate energy surface is found according to a regular dimer method, and final convergence of the dimer is then checked by evaluating accurate energy $E(\mathbf{R}_0)$ and gradient $\nabla E(\mathbf{R}_0)$ at the middle point $\mathbf{R}_0$ of the relaxed dimer. Figure 4.6 shows the progression of the GP-dimer algorithm in the same example task as shown in figure 3.2 for the regular dimer method. In this simple example, the direction of the lowest curvature of the accurate energy surface is found after two initial rotation rounds, and a saddle point close to the correct location is formed on the approximate energy surface based only on the four data points around the start point. After evaluating the accurate energy and gradient at this predicted saddle point, the GP

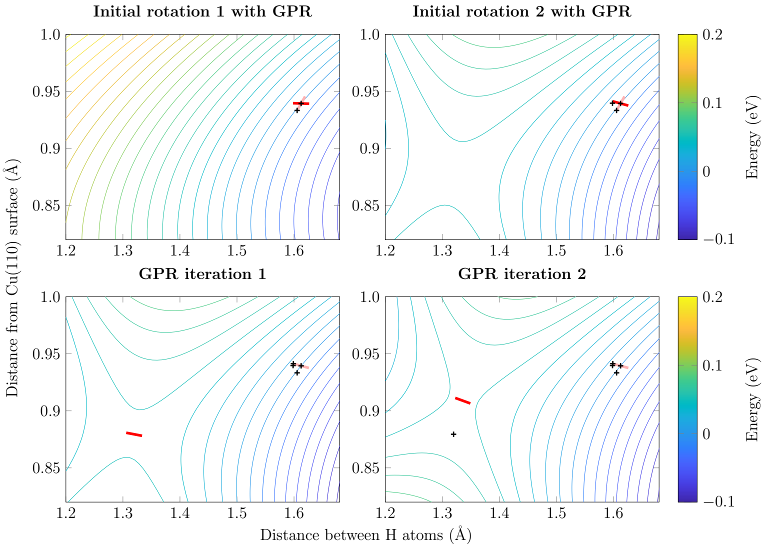


Figure 4.6. Progression of the GP-dimer algorithm in a simple example where two hydrogen atoms are free to move near a fixed copper surface (Mills et al., 1995). The saddle point and the initial dimer coincide with the same two-dimensional cut of the coordinate space as shown in figure 2.2. The pink and red bars represent the dimer in the beginning and end of the initial rotation round or GPR iteration, respectively. During the two rotation rounds, the orientation of the dimer is aligned with the direction of the lowest curvature on an approximate energy surface obtained by GP regression. After each round, accurate energy and its first derivatives are evaluated at one of the images of the dimer, and the information is then added to the training data set (observed locations marked with black crosses). In the actual GPR iterations started after reaching rotational convergence at the start point, the dimer is both rotated and translated to find a saddle point on the approximate energy surface, and final convergence is then checked by evaluating accurate energy and its first derivatives at the middle point of the relaxed dimer. Figure reproduced with permission from Publication IV.

approximation becomes accurate enough for convergence to the correct saddle point.

In addition to the dissociative adsorption of a hydrogen molecule on a copper surface (Mills et al., 1995), used also in GP-NEB calculations in Publication III, the tests of the GP-dimer algorithm in Publication IV involve three gas phase chemical reactions (Birkholtz and Schlegel, 2015) with saddle point configurations illustrated in figure 4.7 alongside the corresponding result graphs. A set of start points is chosen randomly with a varying distance from the saddle point of each example reaction, and the number of energy and gradient evaluations required for convergence is reported for two variants of the regular dimer method and for the GP-dimer algorithm with the inverse-distance covariance function $k_{1/r}$ and stationary squared exponential (k_x) and Matérn-5/2 ($k_x^{M-5/2}$) covariance functions. As shown in figure 4.7, the variants of GP-dimer require fewer evaluations than the regular methods, and the difference increases when

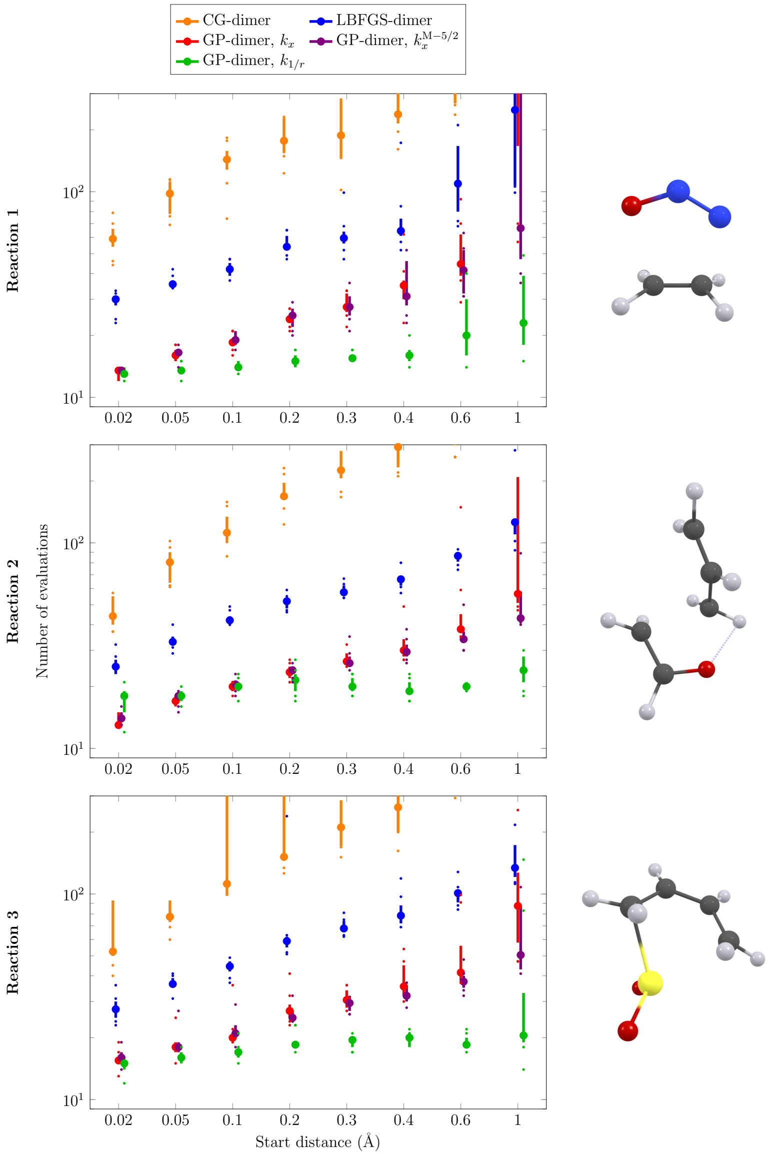


Figure 4.7. Number of energy and gradient evaluations required for convergence with a regular dimer method, based on conjugate gradients (Heyden et al., 2005) or L-BFGS (Kästner and Sherwood, 2008), and the GP-dimer algorithm using the inverse-distance ($k_{1/r}$), squared exponential (k_x), or Matérn-5/2 ($k_x^{\mathrm{M-5/2}}$) covariance function. The saddle point configuration of each of the three example reactions (Birkholtz and Schlegel, 2015) is visualized with the following atom colours: C, dark gray; H, light gray; O, red; N, blue; S, yellow. The distance of the start point from this configuration is shown on the horizontal axis. The large dots present the median number of evaluations among ten randomly chosen start positions, the bars present the interval between the third and eighth largest numbers, and the two smallest and largest numbers are presented with small dots. Figure reproduced with permission from Publication IV.

the start point is farther from the example saddle point. With start points closer than 0.1 Å to the saddle point, only small differences are observed between the three variants of the GP-dimer algorithm, but the benefits of using the inverse-distance covariance function become apparent with larger start distances.

5. Discussion

This dissertation presents the first steps in utilizing Gaussian process regression to enhance saddle point search algorithms on potential energy surfaces. In the GP-NEB algorithm, a minimum energy path between two known minimum energy configurations and a saddle point located at the maximum point of the path are found on an approximate energy surface based on a Gaussian process model, which is updated with accurate observations of energy and its derivatives until convergence of the path on the accurate energy surface is confirmed. In the GP-dimer algorithm, a similar approach is applied to minimum mode following calculations, where only a start point for a saddle point search is given in the beginning. Based on simple test examples, the Gaussian process regression approach may reduce the required number of accurate energy and force evaluations by an order of magnitude when compared with conventional methods.

In Gaussian process regression, the predictions of energy and its derivatives are expressed as probability distributions obtained as a result of Bayesian inference. The variance of the predictive distribution can be utilized in the GP-NEB algorithm as an uncertainty estimate when selecting new observation points from the discretized path. This approach has similarities with Bayesian optimization, where an acquisition function based on the predictive distribution of the objective function is defined for the selection of observation points in a global optimization task (Shahriari et al., 2016). A major difference is that saddle point search algorithms are typically satisfied with a local type of convergence, which means that exploration of uncertain regions far from the predicted minimum energy path or saddle point is not necessary. When the task is to find a minimum energy path with convergence confirmed by accurate evaluations at all points of the discretized path, it is often most efficient to select one of those points as the new observation point. However, if accurate convergence is important only for the saddle point, the convergence of the rest of the path can be defined based on the estimated uncertainty without restricting to any number of discretization points (Garrido Torres et al., 2019). The search for the energy maximum along the path can be then defined as a Bayesian optimization problem with various possible choices for the acquisition function.

Besides algorithmic development, the dissertation shows that automated and accurate modelling of the surroundings of a minimum energy path is possible with rather simple Gaussian process models. While stationary covariance functions with similar properties in all parts of the space of atom coordinates turn out to be insufficient in many systems involving large repulsive forces, a good representation can be obtained by defining the difference measure between two configurations based on inverse interatomic distances. More sophisticated descriptors designed for modelling entire potential energy surfaces are often based on approximation of the total energy of the system with a sum over local atomic environments and may require larger noise variance to be assumed for the observations (Bartók et al., 2010; Bartók and Csányi, 2015). With reduced convergence requirements, however, such models may provide useful properties also for the GP-NEB and GP-dimer algorithms.

The advantage of the Gaussian process regression approach to saddle point searches relies on the assumption that the accurate energy and gradient evaluations are significantly more expensive than predictions based on the Gaussian process model or training of the model. In large systems, however, the applicability of the approach is limited due to the poor scaling of the computational cost of Gaussian process regression with respect to the number of training observations. Since the number of available derivative observations depends on the number of moving atoms, the computational cost increases fast with the system size if full advantage is taken of the derivative information. While many of the attempts to make Gaussian process models more applicable to large data sets rely on approximations, recent development on exact Gaussian process inference is reducing the training cost from cubic to quadratic without compromising the accuracy (Gardner et al., 2018; Wang et al., 2019). This sort of advancement paves the way for further development of efficient saddle point search methods.

Bibliography

Andersen, H. C. (1980). Molecular dynamics simulations at constant pressure and/or temperature. *J. Chem. Phys.*, volume 72, issue 4, pages 2384–2393.

Ásgeirsson, V. and Jónsson, H. (2018). Exploring potential energy surfaces with saddle point searches. In Andreoni, W. and Yip, S. (editors), *Handbook of Materials Modeling: Methods: Theory and Modeling*, Springer: Cham.

Banerjee, A., Adams, N., Simons, J., and Shepard, R. (1985). Search for stationary points on surfaces. *J. Phys. Chem.*, volume 89, issue 1, pages 52–57.

Bartók, A. P. and Csányi, G. (2015). Gaussian approximation potentials: a brief tutorial introduction. *Int. J. Quantum Chem.*, volume 115, issue 16, pages 1051–57.

Bartók, A. P., Payne, M. C., Condor, R., and Csányi, G. (2010). Gaussian approximation potentials: the accuracy of quantum mechanics, without the electrons. *Phys. Rev. Lett.*, volume 104, issue 13, article 136403.

Birkholz, A. B. and Schlegel, H. B. (2015). Using bonding to guide transition state optimization. *J. Comput. Chem.*, volume 36, issue 15, pages 1157–1166.

Bitzek, E., Koskinen, P., Gähler, F., Moseler, M., and Gumbsch, B. (2006). Structural relaxation made simple. *Phys. Rev. Lett.*, volume 97, issue 17, article 170201.

Blight, B. J. N. and Ott, L. (1975). A Bayesian approach to model inadequacy for polynomial regression. *Biometrika*, volume 62, issue 1, pages 79–88.

Bohner, M. U., Meisner, J., and Kästner, J. (2013). A quadratically-converging nudged elastic band optimizer. *J. Chem. Theory Comput.*, volume 9, issue 8, pages 3498–3504.

Cerjan, C. J. and Miller, W. H. (1981). On finding transition states. *J. Chem. Phys.*, volume 75, issue 6, pages 2800–2806.

Chill, S. T., Stevenson, J., Ruhle, V., Shang, C., Xiao, P., Farrell, J. D., Wales, D. J., and Henkelman, G. (2014). Benchmarks for characterization of minima, transition states and pathways in atomic, molecular, and condensed matter systems. *J. Chem. Theory Comput.*, volume 10, issue 12, pages 5476–5482.

Csató, L. and Opper, M. (2002). Sparse on-line Gaussian processes. *Neural Comput.*, volume 14, issue 3, pages 641–668.

Deisenroth, M. P. and Ng, J. W. (2015). Distributed Gaussian processes. In Bach, F. and Blei, D. (editors), *Proceedings of the Thirty-Second International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1481–1490.

Denzel, A. and Kästner, J. (2018a). Gaussian process regression for geometry optimization. *J. Chem. Phys.*, volume 148, issue 9, article 94114.

Denzel, A. and Kästner, J. (2018b). Gaussian process regression for transition state search. *J. Chem. Theory Comput.*, volume 14, issue 11, pages 5777–5786.

E, W., Ren, W., and Vanden-Eijnden, E. (2002). String method for the study of rare events. *Phys. Rev. B*, volume 66, issue 5, article 52301.

E, W., Ren, W., and Vanden-Eijnden, E. (2007). Simplified and improved string method for computing the minimum energy paths in barrier-crossing events. *J. Chem. Phys.*, volume 126, issue 16, article 164103.

Fletcher, R. and Reeves, C. M. (1964). Function minimization by conjugate gradients. *Comput. J.*, volume 7, issue 2, pages 149–154.

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

Garrido Torres, J. A., Jennings, P. C., Hansen, M. H., Boes, J. R., and Bligaard, T. (2019). Low-scaling algorithm for nudged elastic band calculations using a surrogate machine learning model. *Phys. Rev. Lett.*, volume 122, issue 15, article 156001.

Gibbs, M. N. and MacKay, D. J. C. (2000). Variational Gaussian process classifiers. *IEEE Trans. Neural Netw.*, volume 11, issue 6, pages 1458–1464.

Henkelman, G., Jóhannesson, G. H., and Jónsson, H. (2000). Methods for finding saddle points and minimum energy paths. In Schwartz, S. D. (editor), *Theoretical Methods in Condensed Phase Chemistry*, Kluwer Academic: New York, pages 269–300.

Henkelman, G. and Jónsson, H. (1999). A dimer method for finding saddle points on high dimensional potential surfaces using only first derivatives. *J. Chem. Phys.*, volume 111, issue 15, pages 7010–7022.

Henkelman, G. and Jónsson, H. (2000). Improved tangent estimate in the nudged elastic band method for finding minimum energy paths and saddle points. *J. Chem. Phys.*, volume 113, issue 22, pages 9978–9985.

Henkelman, G., Uberuaga, B. P., and Jónsson, H. (2000). A climbing image nudged elastic band method for finding saddle points and minimum energy paths. *J. Chem. Phys.*, volume 113, issue 22, pages 9901–9904.

Hensman, J., Fusi, N., and Lawrence, N. D. (2013). Gaussian processes for big data. In Nicholson, A. and Smyth, P. (editors), *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence*, AUAI Press: Corvallis, pages 282–290.

Heyden, A., Bell, A. T., and Keil, F. J. (2005). Efficient methods for finding transition states in chemical reactions: comparison of improved dimer method and partitioned rational function optimization method. *J. Chem. Phys.*, volume 123, issue 22, article 224101.

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

Kamath, A., Vargas-Hernández, R. A., Krems, R. V., Carrington, T., and Manzhos, S. (2018). Neural networks vs Gaussian process regression for representing potential energy surfaces: a comparative study of fit quality and vibrational spectrum accuracy. *J. Chem. Phys.*, volume 148, issue 24, article 241702.

Kästner, J. and Sherwood, P. (2008). Superlinearly converging dimer method for transition state search. *J. Chem. Phys.*, volume 128, issue 1, article 14106.

Keck, J. C. (1967). Variational theory of reaction rates. In Prigogine, I. (editor), *Advances in Chemical Physics*, volume 13, Wiley: New York, pages 85–121.

Kimeldorf, G. S. and Wahba, G. (1970). A correspondence between Bayesian estimation on stochastic processes and smoothing by splines. *Ann. Math. Stat.*, volume 41, issue 2, pages 495–502.

Kolmogorov, A. N. (1941). Interpolation und extrapolation von stationären zufälligen folgen. *Izv. Akad. Nauk SSSR Ser. Mat.*, volume 5, issue 1, pages 3–14.

Kramers, H. A. (1940). Brownian motion in a field of force and the diffusion model of chemical reactions. *Physica*, volume 7, issue 4, pages 284–304.

Krige, D. G. (1951). A statistical approach to some basic mine valuation problems on the Witwatersrand. *J. South. Afr. Inst. Min. Metall.*, volume 52, issue 6, pages 119–139.

Lampinen, J. and Vehtari, A. (2001). Bayesian approach for neural networks: review and case studies. *Neural Netw.*, volume 14, issue 3, pages 257–274.

Liu, D. C. and Nocedal, J. (1989). On the limited memory BFGS method for large scale optimization. *Math. Program.*, volume 45, issues 1–3, pages 503–528.

Malek, R. and Mousseau, N. (2000). Dynamics of Lennard-Jones clusters: a characterization of the activation-relaxation technique. *Phys. Rev. E*, volume 62, issue 6, pages 7723–7728.

Maras, E., Trushin, O., Stukowski, A., Ala-Nissilä, T., and Jónsson, H. (2016). Global transition path search for dislocation formation in Ge on Si(001). *Comput. Phys. Commun.*, volume 205, pages 13–21.

Matérn, B. (1960). *Spatial Variation*. Allmänna förlaget: Stockholm.

Matheron, G. (1963). Principles of geostatistics. *Econ. Geol.*, volume 58, issue 8, pages 1246–1266.

Mills, G., Jónsson, H., and Schenter, G. K. (1995). Reversible work based transition state theory: application to H_2 dissociative adsorption. *Surf. Sci.*, volume 324, issues 2–3, pages 305–337.

Minka, T. P. (2001). Expectation propagation for approximate Bayesian inference. In Breese, J. S., Koller, D. (editors), *Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence*, Morgan Kaufmann Publishers: San Francisco, pages 362–369.

Mises, R. von (1964). *Mathematical Theory of Probability and Statistics*. Academic Press: New York.

Müller, K. and Brown, L. D. (1979). Location of saddle points and minimum energy paths by a constrained simplex optimization procedure. *Theor. Chim. Acta*, volume 53, issue 1, pages 75–93.

Munro, L. J. and Wales, D. J. (1999). Defect migration in crystalline silicon. *Phys. Rev. B*, volume 59, issue 6, pages 3969–3980.

Neal, R. M. (1995). *Bayesian Learning for Neural Networks*. Ph.D. thesis, University of Toronto.

Neal, R. M. (1999). Regression and classification using Gaussian process priors. In Bernardo, J. M., Berger, J. O., Dawid, A. P., and Smith, A. F. M. (editors), *Bayesian Statistics 6*, Clarendon Press: Oxford, pages 475–501.

Nocedal, J. (1980). Updating quasi-Newton matrices with limited storage. *Math. Comput.*, volume 35, issue 151, pages 773–782.

O'Hagan, A. (1978). Curve fitting and optimal design for prediction. *J. Royal Stat. Soc. B*, volume 40, issue 1, pages 1–42.

O'Hagan, A. (1992). Some Bayesian numerical analysis. In Bernardo, J. M., Berger, J. O., Dawid, A. P., and Smith, A. F. M. (editors), *Bayesian Statistics 4*, Clarendon Press: Oxford, pages 345–363.

Olsen, R. A., Kroes, G. J., Henkelman, G., Arnaldsson, A., and Jónsson, H. (2004). Comparison of methods for finding saddle points without knowledge of the final states. *J. Chem. Phys.*, volume 121, issue 20, pages 9776–9792.

Olver, F. W. J. and Maximon, L. C. (2010). Bessel functions. In Olver, F. W. J., Lozier, D. W., Boisvert, R. F., and Clark, C. W. (editors), *NIST Handbook of Mathematical Functions*, Cambridge University Press: Cambridge, pages 215–286.

Peterson, A. A. (2016). Acceleration of saddle-point searches with machine learning. *J. Chem. Phys.*, volume 145, issue 7, article 74106.

Polak, E. and Ribière, G. (1969). Note sur la convergence de méthodes de directions conjuguées. *ESAIM: Mathematical Modelling and Numerical Analysis – Modélisation Mathématique et Analyse Numérique*, volume 3, issue R1, pages 35–43.

Quiñonero-Candela, J. and Rasmussen, C. E. (2005). A unifying view of sparse approximate Gaussian process regression. *J. Mach. Learn. Res.*, volume 6, pages 1939–1959.

Rasmussen, C. E. (1996). *Evaluations of Gaussian Processes and Other Methods for Non-Linear Regression*. Ph.D. thesis, University of Toronto.

Rasmussen, C. E. (2003). Gaussian processes to speed up hybrid Monte Carlo for expensive Bayesian integrals. In Bernardo, J. M., Dawid, A. P., Berger, J. O., West, M., Heckerman, D., Bayarri, M. J., and Smith, A. F. M. (editors), *Bayesian Statistics 7*, Clarendon Press: Oxford, pages 651–659.

Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian Processes for Machine Learning*. MIT Press: Cambridge.

Ren, W. (2003). Higher order string method for finding minimum energy paths. *Comm. Math. Sci.*, volume 1, issue 2, pages 377–384.

Riihimäki, J. and Vehtari, A. (2010). Gaussian processes with monotonicity information. In Teh, Y. W. and Titterington, M. (editors), *Proceedings of the Thirteenth International Converence on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pages 645–652.

Rue, H., Martino, S., and Chopin, N. (2009). Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations. *J. Royal Stat. Soc. B*, volume 71, issue 2, pages 319–392.

Sansò, F. and Schuh, W.-D. (1987). Finite covariance functions. *Bull. Geodesique*, volume 61, issue 4, pages 331–347.

Seeger, M., Williams, C. K. I., and Lawrence, N. D. (2003). Fast forward selection to speed up sparse Gaussian process regression. In Bishop, C. M. and Frey, B. J. (editors), *Proceedings of the Ninth International Conference on Artificial Intelligence and Statistics*, Society for Artificial Intelligence and Statistics.

Shahriari, B., Swersky, K., Wang, Z., Adams, R. P., and de Freitas, N. (2016). Taking the human out of the loop: a review of Bayesian optimization. *Proc. IEEE*, volume 104, issue 1, pages 148–175.

Sheppard, D., Terrell, R., and Henkelman, G. (2008). Optimization methods for finding minimum energy paths. *J. Chem. Phys.*, volume 128, issue 13, article 134106.

Simons, J., Jørgensen, P., Taylor, H., and Ozment, J. (1983). Walking on potential energy surfaces. *J. Phys. Chem.*, volume 87, issue 15, pages 2745–2753.

Smidstrup, S., Pedersen, A., Stokbro, K., and Jónsson, H. (2014). Improved initial guess for minimum energy path calculations. *J. Chem. Phys.*, volume 140, issue 21, article 214106.

Snelson, E. and Ghahramani, Z. (2006). Sparse Gaussian processes using pseudo-inputs. In Weiss, Y., Schölkopf, B., and Platt, J. C. (editors), *Advances in Neural Information Processing Systems 18*, MIT Press: Cambridge, pages 1257–1264.

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

Stein, M. L. (1999). *Interpolation of Spatial Data*. Springer-Verlag: New York.

Swope, W. C., Andersen, H. C., Berens, P. H., and Wilson, K. R. (1982). 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.*, volume 76, issue 1, pages 637–649.

Titsias, M. K. (2009). Variational learning of inducing variables in sparse Gaussian processes. In van Dyk, D. and Welling, M. (editors), *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics*, volume 5 of *Proceedings of Machine Learning Research*, pages 567–574.

Trygubenko, S. A. and Wales, D. J. (2004). A doubly nudged elastic band method for finding transition states. *J. Chem. Phys.*, volume 120, issue 5, pages 2082–2094.

Vanhatalo, J., Pietiläinen, V., and Vehtari, A. (2010). Approximate inference for disease mapping with sparse Gaussian processes. *Stat. Med.*, volume 29, issue 15, pages 1580–1607.

Vanhatalo, J. and Vehtari, A. (2008). Modelling local and global phenomena with sparse Gaussian processes. In McAllester, D. A. and Myllymäki, P. (editors), *Proceedings of the Twenty-Fourth Conference on Uncertainty in Artificial Intelligence*, AUAI Press: Corvallis, pages 571–578.

Vineyard, G. H. (1957). Frequency factors and isotope effects in solid state processes. *J. Phys. Chem. Solids*, volume 3, issues 1–2, pages 121–127.

Voter, A. F. (1997). Hyperdynamics: accelerated molecular dynamics of infrequent events. *Phys. Rev. Lett.*, volume 78, issue 20, pages 3908–3911.

Wang, K. A., Pleiss, G., Gardner, J. R., Tyree, S., and Wilson, A. G. (2019). Exact Gaussian processes on a million data points. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (editors), *Advances in Neural Information Processing Systems 32*, Curran Associates: Red Hook, pages 14622–14632.

Wiener, N. (1949). *Extrapolation, Interpolation, and Smoothing of Stationary Time Series*. Wiley: New York.

Wigner, E. (1938). The transition state method. *Trans. Faraday Soc.*, volume 34, pages 29–41.

Williams, C. K. I. and Barber, D. (1998). Bayesian classification with Gaussian processes. *IEEE Trans. Pattern Anal. Mach. Intell.*, volume 20, issue 12, pages 1342–1351.

Williams, C. K. I. and Rasmussen, C. E. (1996). Gaussian processes for regression. In Touretzky, D. S., Mozer, M. C., and Hasselmo, M. E. (editors), *Advances in Neural Information Processing Systems 8*, MIT Press: Cambridge, pages 514–520.

Wu, Z. (1995). Compactly supported positive definite radial functions. *Adv. Comput. Math.*, volume 4, issue 1, pages 283–292.

Zhu, X., Thompson, K. C., and Martínez, T. J. (2019). Geodesic interpolation for reaction pathways. *J. Chem. Phys.*, volume 150, issue 16, article 164103.

Errata

Publication I

The early stopping and convergence criteria are described correctly in the Methods section (section 2.3), but there are unfortunate mistakes in the following two sentences in the Results section. A corrected version of Publication I is available as e-print arXiv:1703.10423.

Original sentence (rows 12–15 of section 3.1):

> *The relaxation of the images on this rough estimate of the energy surface does not, however, bring the images too far from the initial placement because of the condition that images cannot be moved in a single iteration by more than a half of the initial distance between the images.*

Corrected sentence:

> *[...] because of the condition that the relaxation phase is stopped early if the convergence measure, i.e., the mean of the magnitudes of the force components perpendicular to the path at the intermediate images, increases.*

Original sentence (rows 3–5 of the caption of figure 4):

> *The convergence tolerance is 0.001 eV/Å for the magnitude of the perpendicular component of the force on any one of the images.*

Corrected sentence:

> *[...] for the mean of the magnitudes of the perpendicular force components at the intermediate images.*

Publications I and II

The computational complexity of one inner iteration of the GP-NEB algorithm is claimed to be linear with respect to the number of degrees of freedom D (row 25 of section 2.3 in Publication I and the last three rows of section IV.A in Publication II). The computational cost of prediction of energy or any gradient component indeed scales linearly with respect to D, but since moving the images requires prediction of the whole gradient vector, the complexity of one inner iteration becomes quadratic with respect to D.

Publication I

Olli-Pekka Koistinen, Emile Maras, Aki Vehtari, and Hannes Jónsson. Minimum energy path calculations with Gaussian process regression. *Nanosystems: Physics, Chemistry, Mathematics*, volume 7, issue 6, pages 925–935, December 2016.

Minimum energy path calculations with Gaussian process regression

O-P. Koistinen[1], E. Maras[2], A. Vehtari[1], H. Jónsson[2,3]

[1]Helsinki Institute for Information Technology HIIT,
Department of Computer Science, Aalto University, Finland

[2]Department of Applied Physics, Aalto University, Finland

[3]Faculty of Physical Sciences, University of Iceland, 107 Reykjavík, Iceland

hj@hi.is

DOI 10.17586/2220-8054-2016-7-6-925-935

The calculation of minimum energy paths for transitions such as atomic and/or spin rearrangements is an important task in many contexts and can often be used to determine the mechanism and rate of transitions. An important challenge is to reduce the computational effort in such calculations, especially when *ab initio* or electron density functional calculations are used to evaluate the energy since they can require large computational effort. Gaussian process regression is used here to reduce significantly the number of energy evaluations needed to find minimum energy paths of atomic rearrangements. By using results of previous calculations to construct an approximate energy surface and then converge to the minimum energy path on that surface in each Gaussian process iteration, the number of energy evaluations is reduced significantly as compared with regular nudged elastic band calculations. For a test problem involving rearrangements of a heptamer island on a crystal surface, the number of energy evaluations is reduced to less than a fifth. The scaling of the computational effort with the number of degrees of freedom as well as various possible further improvements to this approach are discussed.

Keywords: minimum energy path, machine learning, Gaussian process, transition mechanism, saddle point.

Received: 2 December 2016

1. Introduction

The task of predicting the rate and identifying the mechanism of transitions involving some rearrangements of atoms in or on the surface of solids shows up in many different applications, for example diffusion, crystal growth, chemical catalysis, nanotechnology, etc. At a finite temperature, the thermal fluctuations in the dynamics of atoms can lead to rearrangements from one stable configuration to another, but these are rare events on the time scale of atomic vibrations, so direct dynamics simulations cannot in most cases be used for these types of studies. The separation of time scales typically amounts to several orders of magnitude and a direct simulation would take impossibly long time. Instead, algorithms based on statistical mechanics as well as classical dynamics and focusing on the relevant rare events need to be applied [1–3]. Typical transitions involve not just one or a few atoms but rather a large number of atoms so the challenge is also to deal with multiple degrees of freedom. One way of looking at the problem is to characterise the motion of the system on a high dimensional energy surface where the number of degrees of freedom is easily more than a hundred. A key concept is the reaction coordinate which usually is taken to be a minimum energy path (MEP) on the energy surface connecting one minimum to another. The rate of transitions in solids is usually evaluated within harmonic transition state theory which is based on a quadratic expansion of the energy surface at the initial state minimum and at the highest maximum along the MEP, which is a first order saddle point on the energy surface [4]. For given initial and final states, the task is to determine the MEP and identify the saddle point(s) as well as possible unknown, intermediate minima [5]. The discussion here has been in terms of rearrangements of atoms, but similar considerations apply to reorientations of magnetic moments [6–9].

The nudged elastic band (NEB) method is commonly used to find MEPs for atomic rearrangements [5,10,11]. An analogous method, referred to as the geodesic NEB, has been developed for magnetic transitions [12]. In NEB calculations, some initial path is constructed between two local minima on the energy surface and the path is represented by a discrete set of replicas of the system. The replicas are referred to as images of the system. They consist of some set of values for all degrees of freedom in the system. The NEB algorithm then optimises iteratively the location of the images that are between the endpoint minima so as to obtain a discrete representation of the MEP. Initially, the method was mainly used in combination with analytical potential energy functions, but today the method is used extensively in combination with electronic structure calculations. A large amount of computer time is used in these calculations. Each calculation typically involves 100 evaluations of the energy and force (the negative gradient of the energy) for each one of the images and the path is typically represented by 5 to 10 images. Since a typical electronic structure calculation takes on the order of tens of CPU minutes or more, these calculations can be heavy. Also, several different possible final states usually need to be tested and the NEB calculation therefore repeated. In light of the widespread use and large amount of CPU time used in NEB calculations, it is of great practical importance to find ways to accelerate the calculations. The goal should be to use the information coming from all the computationally intensive electronic structure calculations in an optimal way so as to reduce as much as possible the number of iterations needed to reach the MEP.

It has recently been shown that a machine learning algorithm based on neural networks can be used to significantly reduce the computational effort in NEB calculations [13]. An approximate representation of the energy surface is constructed from the calculations using a machine learning approach and the MEP calculated using the NEB method on this approximate surface. Then, additional evaluations are made of the true energy surface, the approximate model surface refined, etc., until convergence on the MEP of the true energy surface has been reached. The number of function evaluations was shown to drop dramatically by applying such an approach [13].

We present here an initial step in the development of a similar approach to accelerated MEP calculations based on Gaussian process regresssion [14–17]. This approach could have some advantages over neural networks for such applications. Neural networks have a large number of weights which can have multimodal distributions making the search for global optimum difficult and leading to possible dependence on the initial values of the parameters [13]. Also, the handling of uncertainties in GP theory is easier than in neural networks since the prediction equations are analytical and integration over the parameter space can be carried out more easily. It is, therefore, of interest to test the efficiency of the GP approach in MEP calculations. We report in this article initial feasability studies. More extensive testing and comparison with other approaches such as neural networks is left for future work.

The article is organized in the following way: The methodology is presented in the next section, followed by a section on applications, both a simple two-dimensional system and a larger test problem involving rearrangements of a heptamer island on a crystal surface. The article concludes with a discussion section.

2. Methods

The method presented here for finding the minimum energy paths can be viewed as an acceleration of a NEB calculation by making use of Gaussian process theory. Previously calculated data points are used to construct an approximate model of the energy surface and the MEP is found for this approximate surface before additional calculations of the true energy are carried out. This gives an interpolation between the calculated points and also provides an extrapolation that can be used to explore the energy surface with larger moves. The savings in computational effort are based on the fact that several computationally light iterations can be made for the approximate surface in between the computationally demanding evaluations of the true energy function. A brief review of the NEB method is first given, then a description of the Gaussian process regression, and finally a detailed algorithm describing how the calculations were carried out in the present case.

2.1. Nudged elastic band method

Given two local minima on the energy surface, the task is to find an MEP connecting the two. The definition of an MEP is that the gradient has zero component perpendicular to the path tangent at each point along the path. The NEB method needs to be started with some initial path between the two minima that is represented by a set of images. Most often, a straight line interpolation between the minima is used to generate the initial path [11], but a better approach is to start with a path that interpolates as closely as possible the changes distances between atoms [18].

The key aspect of the NEB algorithm is the nudging, a force projection which is used to decouple the displacements of the images perpendicular to the path towards the MEP from the displacements that affect their distribution along the path. In order to make this projection, an estimate of the local tangent to the path at each of the images is needed. A numerically stable choice involves finding the line segment from the current image to the adjacent image of higher energy [19].

Given this decoupling, there are several different options for distributing the images along the path. Some constraint is needed to prevent the images from sliding down to the minima at the two ends. In most cases an even distribution is chosen, but one can also choose to have, for example, higher density of images where the energy is larger [20]. An attractive spring force is typically introduced between adjacent images to control the spacing between images and this also prevents the path from becoming arbitrarily long in regions of little or no force. The latter is important, for example, in calculations of adsorption and desorption of molecules at surfaces. For systems that can freely translate and rotate, such as nano-clusters in free space, it is important to remove the translational and rotational degrees of freedom. This is non-trivial because the system cannot be treated as a rigid body. A method for doing this efficiently based on quaternions has recently been presented [21].

The component of the force acting on each image perpendicular to the path is used to iteratively move the images from the initial path to the MEP. The force is the negative of the gradient and in most cases an evaluation of the energy delivers also the gradient vector at little or no extra expense. The largest amount of information from an evaluation of a point on the energy surface is, therefore, represented by the gradient. It is hower typically too expensive to evaluate second derivatives of the energy and iterative algorithms for moving the images towards the MEP are therefore based solely on the gradient and the energy at each point. A simple and numerically stable method that has been used extensively in NEB calculations will be used here. It is based on the velocity Verlet method where only the component of the velocity in the direction of the force is included and the velocity is zered of the its dot product with the force becomes negative [11]. A somewhat higher efficiency can be obtained by using a quadratically convergent algorithm such as conjugate gradients or quasi-Newton [22] but those can be less stable especially in the beginning of an NEB calculation. A linear interpolation between the initial state minima was used in all the calculations presented here and the number of images, N_p, chosen to be either 5 or 8. An equal distribution of the images along the path was chosen.

The focus here is on calculations where the energy and the gradient are obtained using some ab initio or density functional theory calculation. The computational effort in all other parts of the calculation is then insignificant in comparison and the computational effort is well characterised by simply the number of times the energy and force need to be evaluated in order to converge on the MEP. Below, we introduce a strategy to accelerate the MEP search with Gaussian process regression.

2.2. Gaussian processes regression

The general idea behind the strategy is similar to the one introduced by Peterson [13]. The idea is to use the calculations carried out so far to train an approximate model of the energy surface, and find the MEP with the conventional methods using the approximations of the energy and gradient based on this model. After converging to the MEP on the approximate energy surface, the true energy and force are evaluated again, showing whether or not the path has converged to the true MEP. If not, the model is updated with the new values of the true energy and force to get a more accurate approximation, and this is continued iteratively, until the true MEP has been found. Since the number of true energy and force evaluations is the measure of computational effort, basically any method can be used to optimise the path on the approximate energy surface, as long as it converges to an MEP.

Here, a Gaussian process (GP) is used as a probabilistic model for the energy surface. GPs provide a flexible framework for modelling multidimensional functions. Through the selection of the covariance function and its hyperparameters, smoothness properties of the function can easily be defined and those properties can also be learned from the data. It is also straightforward to both include derivative observations into the model and to predict the derivative of the modelled function. Analytical expressions for the posterior predictions conditional on the hyperparameters allow both fast predictions and reliable estimation of uncertainties. In cases where only a small number of observations are available, Gaussian processes have been shown to have good predictive performance compared to other machine learning methods [23].

A GP can be seen as a probability distribution over functions in a continuous domain, see, e.g., [14–17]. In a GP, the joint probability distribution of the function values $f(\mathbf{x}^{(1)}), f(\mathbf{x}^{(2)}), \ldots, f(\mathbf{x}^{(N)})$ at any finite set of input points $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(N)} \in \mathbb{R}^D$ is a multivariate Gaussian distribution. A GP is defined by a mean function $m(\mathbf{x})$ and a covariance function $k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)})$, which determines the covariance between $f(\mathbf{x}^{(i)})$ and $f(\mathbf{x}^{(j)})$, e.g., based on the distance between $\mathbf{x}^{(i)}$ and $\mathbf{x}^{(j)}$.

Consider a regression problem $y = f(\mathbf{x}) + \epsilon$, where ϵ is Gaussian noise with variance σ^2, and a training data set $\{\mathbf{X}, \mathbf{y}\}$, where $\mathbf{X} \in \mathbb{R}^{N \times D}$ denotes a matrix of N input vectors $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(N)} \in \mathbb{R}^D$ and $\mathbf{y}$ is a vector of the corresponding N noisy observations. By choosing a Gaussian process to model function f, different prior assumptions can be made about the properties of the function, and after observing $\{\mathbf{X}, \mathbf{y}\}$, the posterior predictive probabilities for the function values at a set of new points can be calculated analytically as a multivariate Gaussian distribution. Here, the mean function is taken to be $m(\mathbf{x}) = 0$ and the covariance function is assumed to have the form

$$k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = c^2 + \eta^2 \exp\left(-\frac{1}{2}\sum_{d=1}^{D} \rho_d^{-2}(x_d^{(i)} - x_d^{(j)})^2\right),$$

where η^2 and $\boldsymbol{\rho} = \{\rho_1, \ldots, \rho_D\}$ are the hyperparameters of the GP model. The squared exponential covariance function is infinitely differentiable and thus favours smooth functions. The length scales $\boldsymbol{\rho}$ define how fast the function f can change, and η^2 controls the magnitude of the overall variation. The additional constant term c^2 has a similar effect as integration over an unknown constant mean function with a Gaussian prior distribution of variance of c^2. The posterior predictive distribution for a function value of the function at a new point $\mathbf{x}^*$, denoted as f^*, is described by a Gaussian distribution with mean

$$E[f^*|\mathbf{x}^*, \mathbf{y}, \mathbf{X}, \boldsymbol{\theta}] = K(\mathbf{x}^*, \mathbf{X})(K(\mathbf{X}, \mathbf{X}) + \sigma^2\mathbf{I})^{-1}\mathbf{y}$$

and variance

$$\mathrm{Var}[f^*|\mathbf{x}^*, \mathbf{y}, \mathbf{X}, \boldsymbol{\theta}] = k(\mathbf{x}^*, \mathbf{x}^*) - K(\mathbf{x}^*, \mathbf{X})(K(\mathbf{X}, \mathbf{X}) + \sigma^2\mathbf{I})^{-1}K(\mathbf{X}, \mathbf{x}^*),$$

where $\mathbf{I}$ is the identity matrix and the notation $K(\mathbf{X}, \mathbf{X}')$ represents a covariance matrix with entries $K_{ij} = k(\mathbf{x}^{(i)}, \mathbf{x}'^{(j)})$. The hyperparameter values $\boldsymbol{\theta} = \{\eta^2, \boldsymbol{\rho}\}$ are optimised by defining a prior probability distribution $p(\boldsymbol{\theta})$ and maximising the marginal posterior probability $p(\boldsymbol{\theta}|\mathbf{y}, \mathbf{X}) = p(\boldsymbol{\theta})p(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta})$ after observing $\mathbf{y}$.

Since differentiation is a linear operation, the derivative of a Gaussian process is also a Gaussian process (see, e.g., [24, 25]), and this makes it possible to use observations of the derivative of the function and also to predict derivatives of the function f. The partial derivative observations can simply be included in the observation vector $\mathbf{y}$ and the covariance matrix correspondingly extended with the covariances between the observations and the partial derivatives and the covariances between the partial derivatives themselves. In the case of the squared exponential covariance function, these entries are obtained by

$$\begin{aligned}\mathrm{Cov}\left[\frac{\partial f^{(i)}}{\partial x_d^{(i)}}, f^{(j)}\right] =& \frac{\partial}{\partial x_d^{(i)}}\mathrm{Cov}\left[f^{(i)}, f^{(j)}\right] = \frac{\partial}{\partial x_d^{(i)}} k\left(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}\right) \\ =&\eta^2 \exp\left(-\frac{1}{2}\sum_{g=1}^{D} \rho_g^{-2}(x_g^{(i)} - x_g^{(j)})^2\right)\left(-\rho_d^{-2}(x_d^{(i)} - x_d^{(j)})\right),\end{aligned}$$

and

$$\begin{aligned}\mathrm{Cov}\left[\frac{\partial f^{(i)}}{\partial x_{d_1}^{(i)}}, \frac{\partial f^{(j)}}{\partial x_{d_2}^{(j)}}\right] =& \frac{\partial^2}{\partial x_{d_1}^{(i)}\partial x_{d_2}^{(j)}}\mathrm{Cov}\left[f^{(i)}, f^{(j)}\right] = \frac{\partial^2}{\partial x_{d_1}^{(i)}\partial x_{d_2}^{(j)}} k\left(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}\right) \\ =&\eta^2 \exp\left(-\frac{1}{2}\sum_{g=1}^{D} \rho_g^{-2}(x_g^{(i)} - x_g^{(j)})^2\right) \times \\ &\rho_{d_1}^{-2}\left(\delta_{d_1 d_2} - \rho_{d_2}^{-2}(x_{d_1}^{(i)} - x_{d_1}^{(j)})(x_{d_2}^{(i)} - x_{d_2}^{(j)})\right),\end{aligned}$$

where $\delta_{d_1 d_2} = 1$ if $d_1 = d_2$, and $\delta_{d_1 d_2} = 0$ if $d_1 \neq d_2$.

These same expressions are useful also when predicting values of the derivatives. The posterior predictive distribution of the partial derivative of function f with respect to dimension d at a new point $\mathbf{x}^*$ is a Gaussian distribution with mean

$$E\left[\frac{\partial f^*}{\partial x_d^*}\middle|\mathbf{x}^*, \mathbf{y}, \mathbf{X}, \boldsymbol{\theta}\right] = \frac{\partial K(\mathbf{x}^*, \mathbf{X})}{\partial x_d^*}(K(\mathbf{X}, \mathbf{X}) + \sigma^2\mathbf{I})^{-1}\mathbf{y}$$

and variance

$$\mathrm{Var}\left[\frac{\partial f^*}{\partial x_d^*}\middle|\mathbf{x}^*, \mathbf{y}, \mathbf{X}, \boldsymbol{\theta}\right] = \frac{\partial^2 k(\mathbf{x}^*, \mathbf{x}^*)}{\partial x_d^* \partial x_d^*} - \frac{\partial K\left(\mathbf{x}^*, \mathbf{X}\right)}{\partial x_d^*}(K(\mathbf{X}, \mathbf{X}) + \sigma^2\mathbf{I})^{-1}\frac{\partial K(\mathbf{X}, \mathbf{x}^*)}{\partial x_d^*}.$$

In the present application, the vector $\mathbf{x}$ represents coordinates of the atoms and the function f the energy of the system. The observations $\mathbf{y}$ are the true values of the energy as well as the partial derivatives of the energy with respect to the coordinates of the atoms at the various sets of coordinates $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(N)}$. With this input, the Gaussian process model is used to predict the most likely value of energy f^* and its derivatives $\frac{\partial f^*}{\partial x_d^*}$ at a new set of atom coordinates $\mathbf{x}^*$ representing in this case an image in the discrete path representation between the initial and final state minima. Since the training data is assumed to be noiseless and include also derivative observations, the equations for the mean predictions can be presented as

$$E[f^*|\mathbf{x}^*, \mathbf{y}_{ext}, \mathbf{X}, \boldsymbol{\theta}] = K_{ext}^* K_{ext}^{-1}\mathbf{y}_{ext} \tag{1}$$

and

$$E\left[\frac{\partial f^*}{\partial x_d^*}\middle|\mathbf{x}^*, \mathbf{y}_{ext}, \mathbf{X}, \boldsymbol{\theta}\right] = \frac{\partial K_{ext}^*}{\partial x_d^*} K_{ext}^{-1}\mathbf{y}_{ext}, \tag{2}$$

where

$$\mathbf{y}_{ext}^* = \left[y^{(1)} \cdots y^{(N)}, \frac{\partial f^{(1)}}{\partial x_1^{(1)}} \cdots \frac{\partial f^{(N)}}{\partial x_1^{(N)}}, \frac{\partial f^{(1)}}{\partial x_2^{(1)}} \cdots \frac{\partial f^{(N)}}{\partial x_2^{(N)}}, \quad \cdots \quad, \frac{\partial f^{(1)}}{\partial x_D^{(1)}} \cdots \frac{\partial f^{(N)}}{\partial x_D^{(N)}}\right]^T,$$

$$K^*_{ext} = \left[K(\mathbf{x}^*, \mathbf{X}) \quad \frac{\partial K(\mathbf{x}^*, \mathbf{X})}{\partial x_1} \quad \frac{\partial K(\mathbf{x}^*, \mathbf{X})}{\partial x_2} \quad \cdots \quad \frac{\partial K(\mathbf{x}^*, \mathbf{X})}{\partial x_D} \right],$$

and

$$K_{ext} = \begin{bmatrix} K(\mathbf{X}, \mathbf{X}) & \frac{\partial K(\mathbf{X}, \mathbf{X}')}{\partial x'_1} & \frac{\partial K(\mathbf{X}, \mathbf{X}')}{\partial x'_2} & \cdots & \frac{\partial K(\mathbf{X}, \mathbf{X}')}{\partial x'_D} \\ \frac{\partial K(\mathbf{X}, \mathbf{X}')}{\partial x_1} & \frac{\partial^2 K(\mathbf{X}, \mathbf{X}')}{\partial x_1 x'_1} & \frac{\partial^2 K(\mathbf{X}, \mathbf{X}')}{\partial x_1 x'_2} & \cdots & \frac{\partial^2 K(\mathbf{X}, \mathbf{X}')}{\partial x_1 x'_D} \\ \frac{\partial K(\mathbf{X}, \mathbf{X}')}{\partial x_2} & \frac{\partial^2 K(\mathbf{X}, \mathbf{X}')}{\partial x_2 x'_1} & \frac{\partial^2 K(\mathbf{X}, \mathbf{X}')}{\partial x_2 x'_2} & \cdots & \frac{\partial^2 K(\mathbf{X}, \mathbf{X}')}{\partial x_2 x'_D} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial K(\mathbf{X}, \mathbf{X}')}{\partial x_D} & \frac{\partial^2 K(\mathbf{X}, \mathbf{X}')}{\partial x_D x'_1} & \frac{\partial^2 K(\mathbf{X}, \mathbf{X}')}{\partial x_D x'_2} & \cdots & \frac{\partial^2 K(\mathbf{X}, \mathbf{X}')}{\partial x_D x'_D} \end{bmatrix}.$$

2.3. Algorithm for GP-aided MEP search

```
Input: the coordinates, energy and its gradient at the two minima on the energy surface, the number of images representing the path ($N_p$), convergence limit ($CL$), step coefficient ($k_{step}$).
Output: minimum energy path represented by $N_p$ images.
1. Place the initial $N_p$ images equally spaced along a straight line between the two minima.
2. Repeat until convergence (outer iteration loop):
    A. Evaluate the true energy and its gradient at the $N_p - 2$ intermediate images of the path, and add them to the training data.
    B. Calculate the negative energy gradient (e.g., force) component perpendicular to the path ($ngc$) for each intermediate image, and denote the mean of their norms as $M_{ngc}$.
    C. If $M_{ngc} < CL$, the path has converged to the true MEP.
    D. Optimise the hyperparameters of the GP model based on the training data, and calculate the matrix inversion in equation 1.
    E. Define $CL_{relax}$ as $\frac{1}{10}$ of the smallest $M_{ngc}$ so far, and repeat (relaxation phase):
        I. Move the intermediate images according to any stable path optimisation algorithm.
        II. Update the GP posterior mean energy and gradient at the new intermediate images using equations 1 and 2.
        III. Calculate $ngc$ for each image using the GP posterior mean gradient, and denote the mean of their norms as $M^{GP}_{ngc}$.
        IV. If $M^{GP}_{ngc} < CL_{relax}$, or if $M^{GP}_{ngc}$ is increasing, exit the relaxation phase (E).
```

The GP calculations make use of the GPStuff toolbox [26]. For the hyperparameter optimisation which is carried out after each evaluation of the true energy and force, the computational effort scales as $\mathcal{O}((N(D+1))^3)$, where N is the number of observations and D is the number of degrees of freedom (here coordinate of movable atoms). Since the hyperparameters and observations stay the same during a search for the MEP on the approximate energy surface, the matrix inversion in equation 1 needs to be computed only once for each such optimization of the path. Thus, the complexity of one inner iteration on the GP posterior energy surface is $\mathcal{O}(N(D+1))$.

The length of any one displacement of an image is restricted to be less than half of the initial interval between the images in order to prevent the path from forming loops. Convergence of the path to the MEP is determined from the norm of the force component perpendicular to the path at each of the intermediate images. The path is considered to be converged to the MEP, when the mean of the true values of these norms is less than 0.001 eV/Å. During the relaxations, norms based on the current GP model are monitored and the mean of these used as a convergence criterion. Since it is not necessary to find a path that is accurately converged on the MEP of the inaccurate, approximate energy surface, the convergence limit for each relaxation phase is defined as $\frac{1}{10}$ of the smallest true mean of norms evaluated so far. Higher convergence limits at early relaxation steps speed up the algorithm and they also make it more stable by preventing the path from escaping too far from the true observation points. For the same reason, the relaxation is stopped before convergence if the convergence criterion starts to increase.

3. Applications

The method described above has been applied to two test problems: A simple two-dimensional problem where the energy surface can be visualised, and a more realistic problem involving the rearrangements of atoms in a heptamer island on a crystal surface.

3.1. Two-dimensional test problem

The two-dimensional problem is formulated by coupling a degree of freedom representing the simultaneous formation and breaking of chemical bonds with a degree of freedom representing a harmonic oscillator solvent environment. The model along with the detailed equations is described in the appendix A.2 of reference [11]. Here, one additional repulsive Gaussian was added to shift the saddle point away from the straight line interpolation between the two minima. A contour graph of the energy surface is shown in Fig. 1.

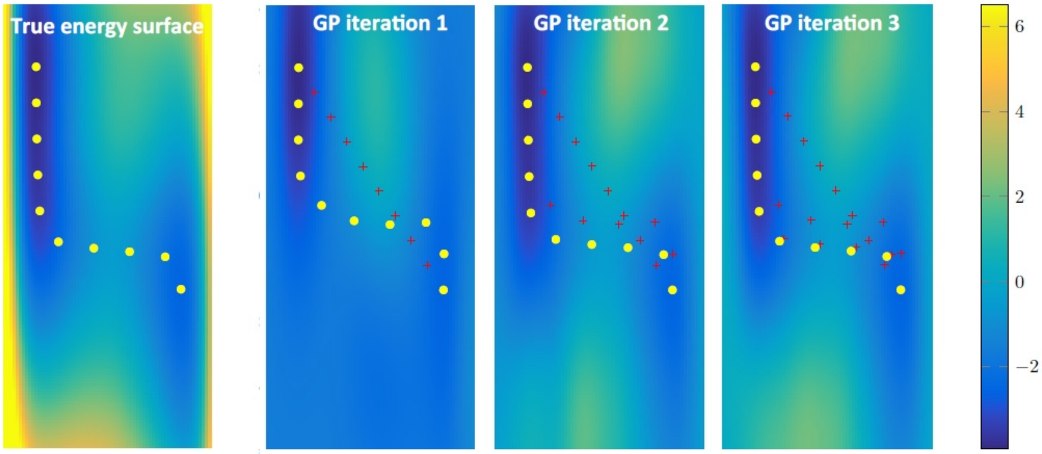


FIG. 1. The true and Gaussian process approximated energy surface and minimum energy path for a two-dimensional test problem. Far left: The true energy surface and points on the minimum energy path (yellow dots). Far right and intermediate figures: The approximate energy surface generated by the Gaussian process regression after one, two and three iterations, points ('images') on the estimated minimum energy path and points where the true energy and force have been calculated (red + signs) at each stage of the calculation the period is missing.

This example shows how the GP model of the energy surface is gradually built up and refined as more observations, i.e. calculations of the true energy and partial derivatives of the energy, are made. Here, $N_p = 10$ images are used to represent the path and the calculation is started by placing the images along a straight line between the two minima on the energy surface. The first observations are made at those points (see red + signs on the figure second from the left). Based on the energy and partial derivatives of the energy at those points, the GP model already shows some of the most important features of the energy surface close to the linear interpolation, but completely misses the increase in energy in the lower half of the figure. The relaxation of the images on this rough estimate of the energy surface does not, however, bring the images too far from the initial placement because of the condition that images cannot be moved in a single iteration by more than half the initial distance between the images. In the second GP iteration, observations are made at the position of the images at the end of the first GP iteration. When those data points are fed into the GP model, the energy surface is already showing the essential features around the MEP, but of course misses the steep increase in the energy far from the MEP. The relaxation of the images during the second GP iteration brings them quite close to the MEP. The addition of observations at those points at the beginning of the third GP iteration refines the model energy surface further. While a a total of six GP iteration are required to bring the images onto the MEP to within the tight tolerance of 0.001 eV/Å in the mean magnitude of the force component perpendicular to the path, no visible changes occur in the contour graph or the location of the images, so the results are not displayed in the figure.

3.2. Heptamer island on a crystal surface

A more realistic test problem which has been used in several studies of MEP and saddle point searches involves an island of 7 atoms on the (111) surface of a face centered cubic (FCC) crystal (see, for example, references [27, 28]). Roughly, this represents a metallic system, but the interaction between the atoms is described here with a simple Morse potential to make it easier to implement the benchmark calculation. The initial, saddle point and final configurations of the atoms for three possible rearrangements of the atoms is shown in Fig. 2. Several other transitions are possible (see reference [27]), but these three are chosen as examples.

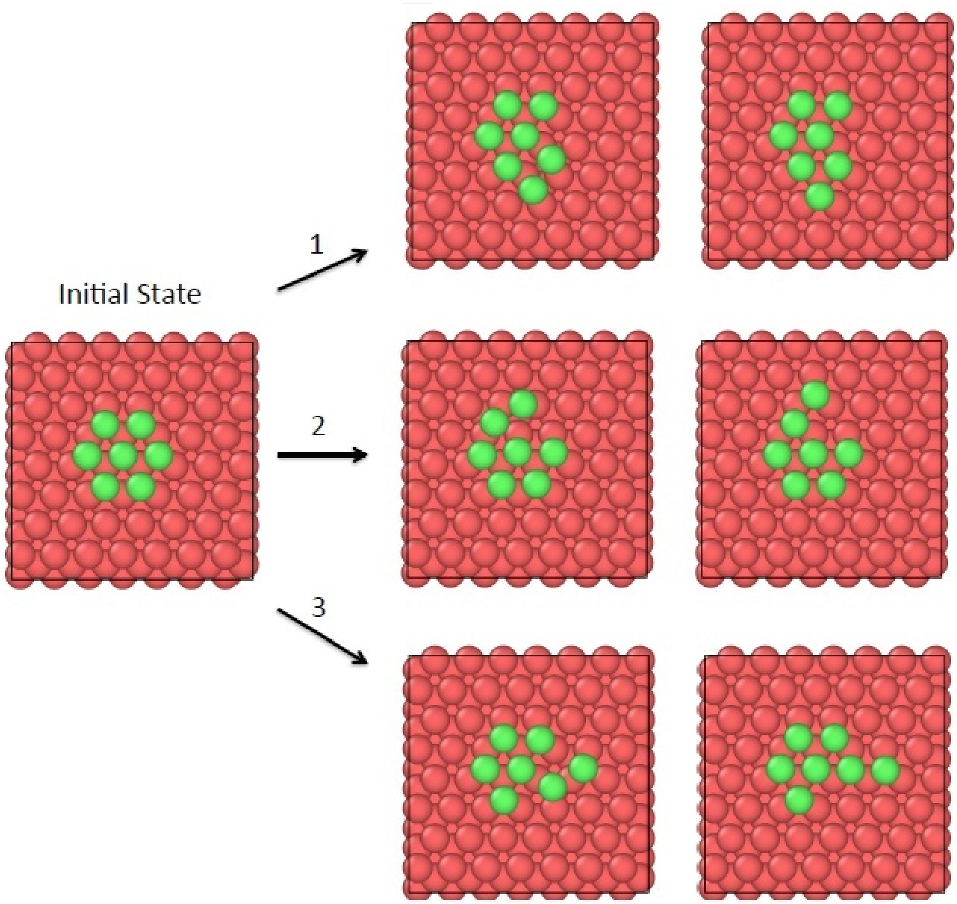


FIG. 2. On-top view of the surface and the seven atom island used to test the efficiency of the Gaussian process regression method. The initial state is shown to the left. The saddle point configurations and the final state configurations of three example transitions are also shown. Transition 1 corresponds to a pair of edge atoms sliding to adjacent FCC sites. In transition 2, an atom half way dissociates from the island. In transition 3, a pair of edge atoms moves in such a way that one of the atoms is displaced away from the island while the other atom takes its place. At the same time the other island atoms as well as some of the underlying atoms also move but in the end return to nearly the same position as they had initially.

The three examples chosen here represent three types of transitions that can occur in the shape of the island. In one case, a pair of edge atoms slides to adjacent FCC sites, in another an atom half way dissociates from the island, and in the third case pair of edge atoms moves in such a way that one of the atoms is displaced away from the island while the other atom takes its place.

The energy along the MEP for the transition 3 is shown in Fig. 3 as well as the energy of the $N_p = 7$ images at the end of GP iterations 1 to 7. After the first and second GP iteration, the estimates of the MEP is quite inaccurate and the energy rises along those paths by more than 3 eV, but already after the third GP iteration, the estimated energy barrier is not too far from the accurate value. After the fifth GP iteration, the shape of the energy curve is quite well reproduced, and after seven iterations the energy along the MEP of the approximate energy surface is nearly indistinguishable from the energy along the true MEP.

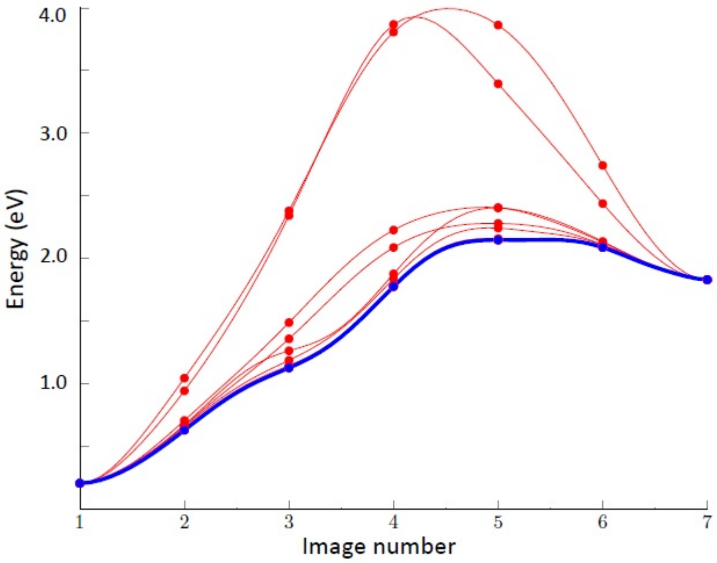


FIG. 3. Energy along paths for transition 3 shown in Fig. 2. The energy of images on the true MEP are shown in blue, but the energy of images on MEPs of approximate models of the energy surface obtained after 1 to 7 Gaussian process iterations are shown in red. After the first two Gaussian process iterations, the energy barrier for this transition is greatly overestimated, but already after three iterations the estimated energy barrier is quite close to the true value, and after 7 iterations an accurate estimate is obtained from the model energy surface.

The number of energy and force evaluations needed to converge the five intermediate images to the MEP in both a regular NEB calculation and in a GP aided calculation was found for varying number of degrees of freedom. The average for the three transitions depicted in Fig. 2 is shown in Fig. 4. The number of degrees of freedom varies from 21 (as only the island atoms are allowed to move while all the substrate atoms are kept immobile), to 42 (as seven of the closest substrate atoms are also allowed to move during the transition). The number of energy and force evaluations for the NEB method obtained here is similar to what has been reported earlier for this test problem, see references [27,28]. It is possible to use a more efficient minimisation scheme to relax the images in NEB calculations [22], but the difference is not large.

A large reduction in the number of energy and force evaluations is obtained by using the GP regression, as shown in Fig. 4. With the GP regression, the reduction is to less than a fifth as compared with the regular NEB calculation. In calculations involving *ab initio* or density functional theory evaluation of the energy and force, the computational effort is essentially proportional to this number of observations and the additional calculations involved in the GP regression is insignificant in comparison. This test problem, therefore, shows that the use of GP regression can significantly reduce the computational effort in, for example, calculations of surface processes.

4. Discussion

The results presented in this article indicate that GP regression is a powerful approach for significantly reducing the computational effort in calculations of MEPs for transitions. This is important since a great deal of computer time is used in such calculations, especially when *ab initio* or density functional theory calculations are used to evaluate the energy and atomic forces. The heptamer island test problem studied here indicates that the

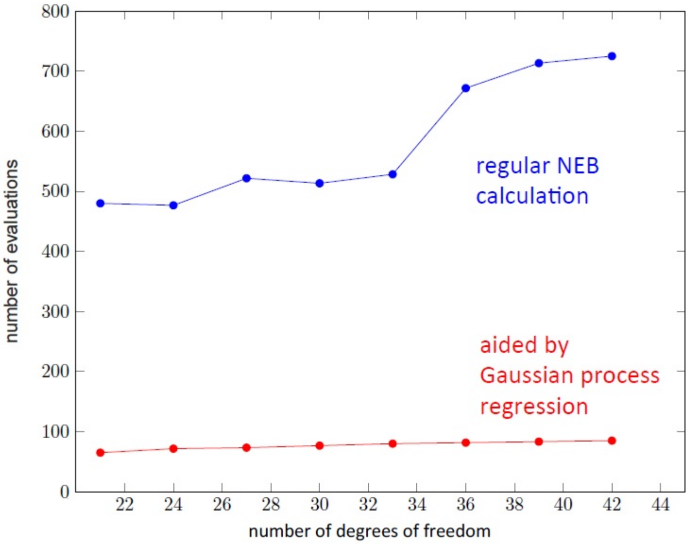


FIG. 4. The average number of energy and force evaluations needed to converge five intermediate images on the minimum energy paths of the three heptamer island transitions shown in Fig. 2 as a function of the number of degrees of freedom included in the calculations. The convergence tolerance is 0.001 eV/Å for the magnitude of the perpendicular component of the force on any one of the images. For the smallest number, 21, only the seven island atoms are allowed to move and all substrate atoms are immobile. For a larger number of degrees of freedom, some of the substrate atoms are also allowed to move during the transition. In the regular NEB calculations (blue dots), the minimization method for relaxing the images to the MEP is based on velocity Verlet algorithm, as described in reference [11]. In the Gaussian process regression calculations (red dots), the number of true energy and function evaluations is less than a fifth of what is needed in the regular NEB calculation. This illustrates well the large reduction in the computational effort that Gaussian process regression can provide in a typical surface process calculation.

computational effort can be reduced to less than a fifth. But, this study represents only an initial proof-of-principle demonstration of the GP regression in this context. There are several ways in which the implementation can be improved and made more efficient. One of the advantages of GP regression over, for example, neural networks is the availability of uncertainty estimates which can be used to make the observations more selective. In the present case, an observation (i.e., evaluation of the true energy and force) was made for all the images in each GP iteration. Alternatively, an observation may only be made for the image for which there is greatest uncertainty. This could target the calculations better and thereby reduce the total number of energy and force evaluations needed to converge to the true MEP.

While the whole path has to be converged well enough to provide an accurate estimate of the tangent, the part of the path that is most important for practical purposes is the region around the first order saddle point. In most cases, the MEP is needed mainly to find the highest energy point along the path, i.e. the first order saddle point on the energy surface that is required for evaluating the transition rate within harmonic transition state theory. The algorithm can be refined to take this into account by, for example, applying the climbing-image NEB [20] where one of the images is driven to the maximum energy along the path, and at the same time the tolerance for the convergence of other images can be increased.

In a typical case, the goal is to evaluate the transition rate using harmonic transition state theory. There, the second derivative matrix, the Hessian matrix, and the frequency of vibrational modes needs to be evaluated at the end points as well as at the (highest) first order saddle point. While the saddle point is not known until the MEP calculation has been carried out, the minima are, and the second derivative matrices at those points might as well be calculated right from the start. This would provide additional information that could be fed into the GP regression so as to improve the accuracy of the approximate energy surface right from first GP iteration. It remains an interesting challenge to extend the GP regression approach to include in some way such information on the second derivatives.

The test problems studied here are quite simple, and it will be important to test the method on more complex systems to fully assess its utility and to develop it further. One issue that can arise is that more than one MEP connects the two endpoint minima. Then, some kind of sampling of MEPs needs to be carried out [29]. Also, some energy surfaces have multiple local minima and highly curved MEPs, which can lead to convergence problems unless a large number of images is included in the calculation. The scaling of the GP regression approach to such more challenging problems needs to be tested. There will, however, clearly be a large set of important problems, such as calculations of catalytic processes, which often involve rather small molecules adsorbed on surfaces, where the complexity is quite similar to the heptamer island test probelm studied here, and where the GP regression is clearly going to offer a significant reduction in computational effort.

At low enough temperature, quantum mechanical tunneling becomes the dominant transition mechanism and the task is then to find the minimum action path [5, 30, 31]. Calculations of tunneling paths requires exploring the energy surface over a wider region than a calculation of MEPs and here again the GP regression approach can lead to a significant reduction in computational effort, even more than for MEP calculations since each iteration necessarily involves more observations and thereby more input for the modeling of the energy surface.

The discussion has focused here on atomic rearrangements, but it will, furthermore, be interesting to apply the GP regression approach to magnetic transitions where the evaluation of the magnetic properties of the system is carried out using computationally intensive *ab initio* or density functional theory calculations. There, the relevant degrees of freedom are the angles defining the orientation of the magnetic vectors and the task is again to find MEPs on the energy surface with respect to those angles [7–9].

Acknowledgements

HJ would like to thank Prof. Andrew Peterson at Brown University for helpful discussions. This work was supported by the Academy of Finland (FiDiPro program grant no. 263294) and by the Icelandic Research Fund.

References

[1] Wigner E. The Transition State Method. *Trans. Faraday Soc.*, 1938, **34**, P. 29.

[2] Kramers H.A. Brownian Motion in a Field of Force and the Diffusion Model of Chemical Reactions. *Physica*, 1940, **7**, P. 284.

[3] Keck J.C. Variational theory of reaction rates. *J. Chem. Phys.*, 1967, **13**, P. 85.

[4] Vineyard G.H. Frequency factors and isotope effects in solid state rate processes. *J. Phys. Chem. Solids*, 1957, **3**, P. 121.

[5] Jónsson H. Simulation of Surface Processes. *Proceedings of the National Academy of Sciences*, 2011, **108**, P. 944.

[6] Bessarab P.F., Uzdin V.M., Jónsson H. Harmonic transition state theory of thermal spin transitions. *Phys. Rev. B*, 2012, **85**, P. 184409.

[7] Bessarab P.F., Uzdin V.M., Jónsson H. Potential Energy Surfaces and Rates of Spin Transitions. *Z. Phys. Chem.*, 2013, **227**, P. 1543.

[8] Bessarab P.F., Uzdin V.M., Jónsson H. Calculations of magnetic states and minimum energy paths of transitions using a noncollinear extension of the Alexander-Anderson model and a magnetic force theorem. *Phys. Rev. B*, 2014, **89**, P. 214424.

[9] Bessarab P.F., Skorodumov A., Uzdin V.M., Jónsson H. Navigation on the energy surface of the noncollinear Alexander-Anderson model. *Nanosystems: Physics, Chemistry, Mathematics*, 2014, **5**, P. 757.

[10] Mills G., Jónsson H., Schenter G.K., Reversible work based transition state theory: Application to H_2 dissociative adsorption. *Surface Science*, 1995, **324**, P. 305.

[11] 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”, edited by B.J. Berne, G. Ciccotti, D.F. Coker, pages 385-404. World Scientific, 1998.

[12] Bessarab P.F., Uzdin V.M., Jónsson H. Method for finding mechanism and activation energy of magnetic transitions, applied to skyrmion and antivortex annihilation. *Comput. Phys. Commun.*, 2015, **196**, P. 335.

[13] Peterson A.A. Acceleration of saddle-point searches with machine learning. *J. Chem. Phys.*, 2016, **145**, P. 074106.

[14] O’Hagan A. Curve fitting and optimal design for prediction. *Journal of the Royal Statistical Society* (Series B), 1978, **40**, P. 1.

[15] MacKay D.J.C. Introduction to Gaussian processes. In “Neural Networks and Machine Learning”, Editor C.M. Bishop, pages 133-166. Springer Verlag, 1998.

[16] Neal R.M. Regression and classification us- ing Gaussian process priors (with discussion). In “Bayesian Statistics 6”, Editors J.M. Bernardo, J.O. Berger, A.P. Dawid, A.F.M. Smith, pages 475-501 (Oxford University Press, 1999).

[17] Rasmussen C.E., Williams C.K.I. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

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

[19] Henkelman G., Jónsson H. Improved Tangent Estimate in the NEB Method for Finding Minimum Energy Paths and Saddle Points. *J. Chem. Phys.*, 2000, **113**, P. 9978.

[20] Henkelman G., Uberuaga B.P., Jónsson H. A Climbing-Image NEB Method for Finding Saddle Points and Minimum Energy Paths. *J. Chem. Phys.*, 2000, **113**, P. 9901.

[21] Melander M., Laasonen K., Jónsson H. Removing external degrees of freedom from transition state search methods using quaternions. *Journal of Chemical Theory and Computation*, 2015, **11**, P. 1055.

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

[23] Lampinen J., Vehtari A. Bayesian approach for neural networks – review and case studies. *Neural Networks*, 2001, **14**, P. 7.

[24] 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”, pages 1033–1040. MIT Press, 2003.

[25] Rasmussen C.E. Gaussian processes to speed up Hybrid Monte Carlo for expensive Bayesian integrals. In “Bayesian Statistics” **7**, pages 651-659. Oxford University Press, 2003.

[26] Vanhatalo J., Riihimäki J., Hartikainen J., Jylänki P., Tolvanen V., Vehtari A. GPstuff: Bayesian Modeling with Gaussian Processes. *Journal of Machine Learning Research*, 2013, **14**, P. 1175.

[27] Henkelman G., Jóhannesson G.H., Jónsson H. Methods for finding saddle points and minimum energy paths. In “Progress in Theoretical Chemistry and Physics”, ed. S. D. Schwartz, Vol. **5**, chapter 10, pages 269-300. Kluwer Academic, Dordrecht, 2000.

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

[29] Maras E., Trushin O., Stukowski A., Ala-Nissila T., Jónsson H. Global transition path search for dislocation formation in Ge on Si(001). *Comput. Phys. Commun.*, 2016, **205**, P. 13.

[30] Mills G., Schenter G.K., Makarov D., Jónsson H. Generalized Path Integral Based Quantum Transition State Theory. *Chemical Physics Letters*, 1997, **278**, P. 91.

[31] Mills G., Schenter G.K., Makarov D. Jónsson H. RAW Quantum Transition State Theory. In “Classical and Quantum Dynamics in Condensed Phase Simulations”, editors B.J. Berne, G. Ciccotti and D.F. Coker, page 405. World Scientific, 1998.

Publication II

Olli-Pekka Koistinen, Freyja B. Dagbjartsdóttir, Vilhjálmur Ásgeirsson, Aki Vehtari, and Hannes Jónsson. Nudged elastic band calculations accelerated with Gaussian process regression. *The Journal of Chemical Physics*, volume 147, issue 15, article 152720, 14 pages, September 2017.

Nudged elastic band calculations accelerated with Gaussian process regression

Olli-Pekka Koistinen,[1,2,3] Freyja B. Dagbjartsdóttir,[2] Vilhjálmur Ásgeirsson,[2] Aki Vehtari,[1] and Hannes Jónsson[2,3,a)]

[1] *Helsinki Institute for Information Technology HIIT, Department of Computer Science, Aalto University, Espoo, Finland*
[2] *Science Institute and Faculty of Physical Sciences, University of Iceland, Reykjavík, Iceland*
[3] *Department of Applied Physics, Aalto University, Espoo, Finland*

(Received 6 June 2017; accepted 29 August 2017; published online 21 September 2017)

Minimum energy paths for transitions such as atomic and/or spin rearrangements in thermalized systems are the transition paths of largest statistical weight. Such paths are frequently calculated using the nudged elastic band method, where an initial path is iteratively shifted to the nearest minimum energy path. The computational effort can be large, especially when *ab initio* or electron density functional calculations are used to evaluate the energy and atomic forces. Here, we show how the number of such evaluations can be reduced by an order of magnitude using a Gaussian process regression approach where an approximate energy surface is generated and refined in each iteration. When the goal is to evaluate the transition rate within harmonic transition state theory, the evaluation of the Hessian matrix at the initial and final state minima can be carried out beforehand and used as input in the minimum energy path calculation, thereby improving stability and reducing the number of iterations needed for convergence. A Gaussian process model also provides an uncertainty estimate for the approximate energy surface, and this can be used to focus the calculations on the lesser-known part of the path, thereby reducing the number of needed energy and force evaluations to a half in the present calculations. The methodology is illustrated using the two-dimensional Müller-Brown potential surface and performance assessed on an established benchmark involving 13 rearrangement transitions of a heptamer island on a solid surface. *Published by AIP Publishing.*
https://doi.org/10.1063/1.4986787

I. INTRODUCTION

Theoretical studies of the transition mechanism and estimation of the rate of thermally activated events involving displacements of atoms or rotations of magnetic moments often involve finding a minimum energy path (MEP) connecting initial and final state minima on the energy surface characterizing the system. An MEP is a natural choice for a reaction coordinate since it represents a path of maximal statistical weight in a system in thermal equilibrium with a heat bath. Transition state theory[1–3] calculations can be carried out using this reaction coordinate to parametrize, for example, a hyperplanar representation of the transition state.[4,5] Even though such a reaction coordinate represents only one particular mechanism for the transition, it is possible to discover a new mechanism corresponding to a lower free energy barrier when full variational optimization of both the location and orientation of the hyperplanar transition state is carried out.[6,7] In such a case, an MEP is just a convenient tool for the implementation of a full free energy calculation.

Most often, transition rates are, however, estimated from the harmonic approximation to transition state theory, where the maximum rise in the energy along an MEP gives the activation energy of the transition and the pre-exponential factor in the Arrhenius expression for the rate can be obtained from the Hessian matrix evaluated at the initial state minimum and the energy maximum—a first-order saddle point on the energy surface.[8] While it is possible to use various methods to converge directly on a saddle point starting from some initial guess, knowledge of the whole MEP is useful because it is important to make sure that the highest first-order saddle point for the full transition has been found. Furthermore, calculations of MEPs often reveal unknown intermediate minima and unexpected transition mechanisms[9] and therefore play an important role in the studies of the mechanism and rate of thermally activated transitions. Most often, such calculations are carried out for transitions involving rearrangements of atoms, but similar considerations apply to thermally activated transitions where magnetic moments rotate from one magnetic state to another.[10–13]

The nudged elastic band (NEB) method is a commonly used iterative approach to find MEPs.[5,9,14] For magnetic transitions, a variant of the method called geodesic NEB has been developed.[15] In the NEB method, the path between two local minima on the energy surface is represented by a discrete set of replicas of the system, referred to as "images," each of them consisting of values for all degrees of freedom. Starting from

[a)] Electronic mail: hj@hi.is

some initial path, the locations of the images on the energy surface are iteratively optimized so as to obtain a discrete representation of an MEP.

Each NEB calculation typically involves on the order of a hundred evaluations of the energy and force (the negative gradient of the energy) for each one of the images, and the path is typically represented by five to ten images. The evaluations were initially performed mostly using analytical potential energy functions, but nowadays electronic structure calculations are also used extensively in NEB applications. Since a typical electronic structure calculation takes on the order of tens of CPU minutes or more, the NEB calculations can become computationally demanding. In addition, the calculation may need to be repeated if there are several possible final states for the transition. Thus, it would be valuable to find ways to accelerate NEB calculations. To get the most out of the computationally intensive electronic structure calculations, the information obtained from them should be exploited better to decrease the number of NEB iterations instead of forgetting it after one iteration.

The use of machine learning to accelerate MEP and saddle point calculations has been introduced by Peterson,[16] who applied neural networks to construct an approximate energy surface for which NEB calculations were carried out. After relaxation of the path on the approximate energy surface, the true energy and force were evaluated at the images of the relaxed path to see whether or not the path had converged on an MEP on the true energy surface. If true convergence had not been reached, the new energy and force values calculated at the location of the images were added to the training data set and the model was updated. This procedure was repeated iteratively until the approximate energy surface was accurate enough for converging on the true MEP.

Proof-of-principle results have also been presented where Gaussian process regression (GPR)[17–20] is applied to accelerate NEB calculations.[21] Since the calculations are largely based on the gradient vector of the energy surface, straightforward inclusion of derivative observations and prediction of derivatives can be seen as advantages of GPR for this application. It is also easy to encode prior assumptions about the smoothness properties of the energy surface into the covariance function of the Gaussian process (GP) model or learn about these properties from the data. Analytical expressions for the posterior predictions conditional on the hyperparameters of the GP model allow both fast predictions and reliable estimation of uncertainties. The predictive performance of GPR has been shown to be competitive with other machine learning methods especially when the number of observations is small.[22]

The GPR approach to MEP calculations is extended here by presenting two algorithms to accelerate climbing image nudged elastic band (CI-NEB) calculations, where one of the images is made to converge to a small tolerance on the highest energy maximum along the MEP.[23] The basic GPR approach is described as the all-images-evaluated (AIE) algorithm, where the energy and force are evaluated at all intermediate images of the CI-NEB before the approximation to the energy surface is updated. In a more advanced algorithm, the energy and force are evaluated at only one image before a new approximate energy surface is constructed. We refer to the latter as the one-image-evaluated (OIE) algorithm. As a probabilistic model, a GP expresses the energy predictions as probability distributions, which means that the uncertainty of the prediction can also be estimated, e.g., as the variance of the posterior distribution. This uncertainty estimate is used by the OIE algorithm to select the image to be evaluated in such a way as to give maximal improvement of the model. By directing the evaluations to locations where they are most needed, the OIE algorithm skips some of the energy and force evaluations and thus decreases the overall computation time compared to the AIE algorithm. This approach has similarities with Bayesian optimization,[24] where the uncertainties of a GP model are used to define an acquisition function that is used to select the locations of new evaluations in a global optimization task.

Another extension of the GPR approach presented here applies when the overall goal is to estimate the forward and backward transition rates using harmonic transition state theory. Then, the Hessian matrix needs to be evaluated at the initial and final state minima, as well as at the highest first-order saddle point along the MEP. The evaluation of the Hessian at the endpoint minima can be carried out before the MEP calculation to provide additional input information into the GP model about the shape of the energy surface in the vicinity of the two ends of the MEP.

The article is organized as follows: In Sec. II, a brief introduction to the NEB method is given, followed by presentation of necessary GP theory for the GPR approach in Sec. III. In Sec. IV, the two implementations, the AIE algorithm and the OIE algorithm, are described and illustrated on the two-dimensional Müller-Brown energy surface.[25] In Sec. V, the heptamer island benchmark is described and performance statistics are given as a function of the number of degrees of freedom. The article concludes with a discussion in Sec. VI.

II. NUDGED ELASTIC BAND METHOD

The objective of the nudged elastic band (NEB) method is to find a minimum energy path (MEP) connecting two given local minima on an energy surface. An MEP is defined as a path for which the gradient of the energy has zero component perpendicular to the path tangent. In the NEB method, the path is represented in a discretized way as a set of images, which are sets of values of all degrees of freedom in the system (atom coordinates and angles specifying orientation of magnetic vectors, and possibly also simulation box size and shape). The MEP is found iteratively, starting from some initial path between the two minima. Most often, a straight line interpolation between the minima has been used to generate the initial path,[14] but a better approach is to start with a path that interpolates as closely as possible the distances between neighboring atoms, the so-called image dependent pair potential (IDPP) method.[26]

The key feature of the NEB algorithm is the "nudging," a projection which is used to separate the force components perpendicular and parallel to the path from each other. If each image is just moved along the force vector (negative gradient of the energy), they would end up sliding down to the nearest minima. The main idea in the NEB method is to take into account only the force component perpendicular to the path and at the same time control the distribution of the images along the path. The projection of the force requires an estimate of the local tangent to the path at the location of each image. A well-behaved estimate is obtained by defining the tangent based on the vector to the neighboring image of higher energy or, if both of the neighbors are either higher or lower in energy than the current image, a weighted average of the vectors to the two neighboring images.[27]

To control the distribution of the images along the path, a spring force acting in the direction of the path tangent is typically introduced. The most common choice is to strive for an even distribution, but one can also choose to have, for example, a higher density of images where the energy is larger.[23] The spring force also prevents the path from becoming arbitrarily long in regions of little or no force. This is important, for example, in calculations of adsorption and desorption of molecules at surfaces.

In each iteration, the images are moved along the resultant vector of the spring force and the component of the true force perpendicular to the path, which is here referred to as the NEB force F_{NEB}. The true force is the negative gradient of the energy, and in most cases, an evaluation of the energy also delivers the gradient vector at little or no additional expense. It is, however, typically too expensive to evaluate the second derivatives of the energy, and the iterative algorithms are therefore based solely on the gradient and energy at each image (in addition to the spring force which depends on the distribution of the images). A simple and stable method that has been used extensively in NEB calculations will be used here to control the step size of the movements. It is based on a velocity Verlet dynamics algorithm where only the component of the velocity in the direction of the NEB force is included as long as the inner product with the NEB force is positive.[14] A somewhat higher efficiency can be obtained by using quadratically convergent algorithms such as conjugate gradient or quasi-Newton,[28] but those can be less stable especially in the beginning of an NEB calculation.

The most important part of an MEP is the vicinity of the highest energy saddle point, especially in harmonic transition state theory calculations where the highest energy saddle point directly gives an estimate of the activation energy of the transition. It is, therefore, advantageous to let the highest energy image move to the maximum energy along the path. This variant of the NEB method is referred to as the climbing image nudged elastic band (CI-NEB) method.[23] Whereas the component of the true force acting in the direction of the tangent is normally removed from the NEB force, for the climbing image, it is instead flipped around to point towards the direction of higher energy along the path. In the CI-NEB method, the spring force is not applied to the climbing image and the rest of the images are distributed evenly on each side of the climbing image. To keep the intervals reasonably similar on both sides of the climbing image, the regular NEB method is typically conducted first (to some preliminary tolerance) so that the image selected as the climbing image is not too far from the true saddle point. The rest of the MEP is mainly needed to ensure that the highest saddle point has been identified and to provide an estimate of the tangent to the path in order to carry out the nudging projections of the forces. It is more important to make the climbing image converge rigorously than the other images. It is, therefore, practical to apply a tighter tolerance for the magnitude of the NEB force acting on the climbing image than to the other images in CI-NEB calculations.

In the heptamer island benchmark presented here, the path was represented by seven images, $N_{\mathrm{im}} = 7$, and the initial path was generated using the IDPP method. All spring constants were chosen to be 1 eV/Å to give an even distribution of the images along the path on each side of the climbing image. The focus here is on calculations where the energy and force are obtained using some *ab initio* or density functional theory calculations. The computational effort in all other parts of the calculation is then insignificant in comparison, and thus the overall computational effort is well characterized by simply the number of times the energy and force need to be evaluated in order to converge on the MEP. Below, we describe various strategies to accelerate CI-NEB calculations with Gaussian process regression.

III. GAUSSIAN PROCESS REGRESSION

A Gaussian process (GP) is a flexible probabilistic model for functions in a continuous domain.[17–20] It is defined by a mean function $m(\mathbf{x})$ and a covariance function $k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)})$ so that 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 $\mathbf{X} = [\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(N)}]^{\mathsf{T}} \in \mathbb{R}^{N\times D}$ 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 with entries $K_{ij} = k(\mathbf{x}_i, \mathbf{x}'_j)$. 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.

The most important part of the GP model is the covariance function, which defines how the function values at any two input points depend on each other, usually based on the distance between the points. Through selection of the covariance function, different prior assumptions about the properties of the function can be encoded into the model. To favor smooth functions, the infinitely differentiable squared exponential covariance function

$$k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = \sigma_{\mathrm{c}}^2 + \sigma_{\mathrm{m}}^2 \exp\left(-\frac{1}{2}\sum_{d=1}^{D}\frac{(x_d^{(i)} - x_d^{(j)})^2}{l_d^2}\right)$$

is used here. The hyperparameters $\boldsymbol{l} = \{l_1, \ldots, l_D\}$ are length scales that define the range of the covariance in each dimension, and σ_{m}^2 is a hyperparameter that controls the magnitude of the covariation. The mean function is here set to zero, but the additional constant term σ_{c}^2 in the covariance function has a similar effect as integration over an unknown constant intercept term having a Gaussian prior distribution with variance σ_{c}^2.

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 N noisy output observations from the N input points $\mathbf{X} = [\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(N)}]^{\mathsf{T}} \in \mathbb{R}^{N \times D}$. The posterior predictive distribution for the function value $f(\mathbf{x}^*)$ at a new point $\mathbf{x}^*$, conditional on the GP model hyperparameters $\boldsymbol{\theta} = \{\sigma_{\mathrm{m}}^2, \boldsymbol{l}\}$, is a Gaussian distribution with mean

$$\mathrm{E}[f(\mathbf{x}^*)|\mathbf{y}, \mathbf{X}, \boldsymbol{\theta}] = K(\mathbf{x}^*, \mathbf{X}) \left(K(\mathbf{X}, \mathbf{X}) + \sigma^2 \mathbf{I}\right)^{-1} \mathbf{y}$$

and variance

$$\begin{aligned}&\mathrm{Var}[f(\mathbf{x}^*)|\mathbf{y}, \mathbf{X}, \boldsymbol{\theta}] \\ &\quad = k(\mathbf{x}^*, \mathbf{x}^*) - K(\mathbf{x}^*, \mathbf{X}) \left(K(\mathbf{X}, \mathbf{X}) + \sigma^2 \mathbf{I}\right)^{-1} K(\mathbf{X}, \mathbf{x}^*),\end{aligned}$$

where $\mathbf{I}$ is the identity matrix. Here the hyperparameter values $\boldsymbol{\theta} = \{\sigma_{\mathrm{m}}^2, \boldsymbol{l}\}$ are 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 $p(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta}) = \int_{\mathbf{f}} p(\mathbf{y}|\mathbf{f}) p(\mathbf{f}|\mathbf{X}, \boldsymbol{\theta}) \mathrm{d}\mathbf{f}$ is the marginal likelihood of $\boldsymbol{\theta}$ in the light of the observed data set $\{\mathbf{X}, \mathbf{y}\}$.

Since differentiation is a linear operation, the derivative of a GP is a GP as well.[29–33] This makes it straightforward to use derivative information and predict derivatives of the function f. Derivative observations can be included in the model by extending the observation vector $\mathbf{y}$ to include partial derivative observations and by extending the covariance matrix $K(\mathbf{X}, \mathbf{X})$ correspondingly to include covariances between the function values and partial derivatives and covariances between the partial derivatives themselves. In the case of the squared exponential covariance function, these entries are obtained by

$$\begin{aligned}&\mathrm{Cov}\left[\frac{\partial f(\mathbf{x}^{(i)})}{\partial x_d^{(i)}}, f(\mathbf{x}^{(j)})\right] \\ &\quad = \frac{\partial}{\partial x_d^{(i)}} \mathrm{Cov}\left[f(\mathbf{x}^{(i)}), f(\mathbf{x}^{(j)})\right] = \frac{\partial k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)})}{\partial x_d^{(i)}} \\ &\quad = -\frac{\sigma_{\mathrm{m}}^2 (x_d^{(i)} - x_d^{(j)})}{l_d^2} \exp\left(-\frac{1}{2} \sum_{g=1}^{D} \frac{(x_g^{(i)} - x_g^{(j)})^2}{l_g^2}\right)\end{aligned}$$

and

$$\begin{aligned}\mathrm{Cov}\left[\frac{\partial f(\mathbf{x}^{(i)})}{\partial x_{d_1}^{(i)}}, \frac{\partial f(\mathbf{x}^{(j)})}{\partial x_{d_2}^{(j)}}\right] &= \frac{\partial^2}{\partial x_{d_1}^{(i)} \partial x_{d_2}^{(j)}} \mathrm{Cov}\left[f(\mathbf{x}^{(i)}), f(\mathbf{x}^{(j)})\right] \\ &= \frac{\partial^2 k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)})}{\partial x_{d_1}^{(i)} \partial x_{d_2}^{(j)}} \\ &= \frac{\sigma_{\mathrm{m}}^2}{l_{d_1}^2} \left(\delta_{d_1 d_2} - \frac{(x_{d_1}^{(i)} - x_{d_1}^{(j)})(x_{d_2}^{(i)} - x_{d_2}^{(j)})}{l_{d_2}^2}\right) \\ &\quad \times \exp\left(-\frac{1}{2} \sum_{g=1}^{D} \frac{(x_g^{(i)} - x_g^{(j)})^2}{l_g^2}\right),\end{aligned}$$

where $\delta_{d_1 d_2} = 1$ if $d_1 = d_2$ and $\delta_{d_1 d_2} = 0$ if $d_1 \neq d_2$. These same expressions are needed when predicting the derivatives. The posterior predictive distribution of the partial derivative of function f with respect to dimension d at a new point $\mathbf{x}^*$ is a Gaussian distribution with mean

$$\mathrm{E}\left[\frac{\partial f(\mathbf{x}^*)}{\partial x_d^*}\middle| \mathbf{y}, \mathbf{X}, \boldsymbol{\theta}\right] = \frac{\partial K(\mathbf{x}^*, \mathbf{X})}{\partial x_d^*} \left(K(\mathbf{X}, \mathbf{X}) + \sigma^2 \mathbf{I}\right)^{-1} \mathbf{y}$$

and variance

$$\mathrm{Var}\left[\frac{\partial f(\mathbf{x}^*)}{\partial x_d^*}\middle| \mathbf{y}, \mathbf{X}, \boldsymbol{\theta}\right] = \frac{\partial^2 k(\mathbf{x}^*, \mathbf{x}^{*\prime})}{\partial x_d^* \partial x_d^{*\prime}} - \frac{\partial K(\mathbf{x}^*, \mathbf{X})}{\partial x_d^*} \left(K(\mathbf{X}, \mathbf{X}) + \sigma^2 \mathbf{I}\right)^{-1} \frac{\partial K(\mathbf{X}, \mathbf{x}^*)}{\partial x_d^*}.$$

In the present application, the vector $\mathbf{x}$ includes coordinates of the atoms and the function f is the energy of the system. The extended observation vector

$$\mathbf{y}_{\mathrm{ext}} = \left[y^{(1)} \cdots y^{(N)}, \frac{\partial f(\mathbf{x}^{(1)})}{\partial x_1^{(1)}} \cdots \frac{\partial f(\mathbf{x}^{(N)})}{\partial x_1^{(N)}}, \frac{\partial f(\mathbf{x}^{(1)})}{\partial x_2^{(1)}} \cdots \frac{\partial f(\mathbf{x}^{(N)})}{\partial x_2^{(N)}}, \ldots, \frac{\partial f(\mathbf{x}^{(1)})}{\partial x_D^{(1)}} \cdots \frac{\partial f(\mathbf{x}^{(N)})}{\partial x_D^{(N)}}\right]^{\mathsf{T}}$$

includes the accurate values of the energy and the partial derivatives of the energy with respect to the coordinates of the atoms (i.e., components of the negative force vector) at the training data points $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(N)}$. The GP model is used to predict the energy $f(\mathbf{x}^*)$ and its gradient vector $[\frac{\partial f(\mathbf{x}^*)}{\partial x_1^*}, \frac{\partial f(\mathbf{x}^*)}{\partial x_2^*}, \ldots, \frac{\partial f(\mathbf{x}^*)}{\partial x_D^*}]^{\mathsf{T}}$ at a new point $\mathbf{x}^*$, which in this case represents an image on the discrete path representation between the initial and final state minima. Since the training data also include derivative observations, the mean and variance of the posterior predictive distribution of the energy are given as

$$\mathrm{E}[f(\mathbf{x}^*)|\mathbf{y}_{\mathrm{ext}}, \mathbf{X}, \boldsymbol{\theta}] = \mathbf{K}_{\mathrm{ext}}^* \left(\mathbf{K}_{\mathrm{ext}} + \sigma^2 \mathbf{I}\right)^{-1} \mathbf{y}_{\mathrm{ext}} \tag{1}$$

and

$$\mathrm{Var}[f(\mathbf{x}^*)|\mathbf{y}_{\mathrm{ext}}, \mathbf{X}, \boldsymbol{\theta}] = k(\mathbf{x}^*, \mathbf{x}^*) - \mathbf{K}_{\mathrm{ext}}^* \left(\mathbf{K}_{\mathrm{ext}} + \sigma^2 \mathbf{I}\right)^{-1} \mathbf{K}_{\mathrm{ext}}^{*\,\mathsf{T}}, \tag{2}$$

where

$$\mathbf{K}_{\mathrm{ext}}^* = \left[K(\mathbf{x}^*, \mathbf{X}) \quad \frac{\partial K(\mathbf{x}^*, \mathbf{X})}{\partial x_1} \quad \frac{\partial K(\mathbf{x}^*, \mathbf{X})}{\partial x_2} \quad \cdots \quad \frac{\partial K(\mathbf{x}^*, \mathbf{X})}{\partial x_D}\right]$$

and

$$\mathbf{K}_{\text{ext}} = \begin{bmatrix} K(\mathbf{X},\mathbf{X}) & \dfrac{\partial K(\mathbf{X},\mathbf{X}')}{\partial x_1'} & \dfrac{\partial K(\mathbf{X},\mathbf{X}')}{\partial x_2'} & \cdots & \dfrac{\partial K(\mathbf{X},\mathbf{X}')}{\partial x_D'} \\ \dfrac{\partial K(\mathbf{X},\mathbf{X}')}{\partial x_1} & \dfrac{\partial^2 K(\mathbf{X},\mathbf{X}')}{\partial x_1 \partial x_1'} & \dfrac{\partial^2 K(\mathbf{X},\mathbf{X}')}{\partial x_1 \partial x_2'} & \cdots & \dfrac{\partial^2 K(\mathbf{X},\mathbf{X}')}{\partial x_1 \partial x_D'} \\ \dfrac{\partial K(\mathbf{X},\mathbf{X}')}{\partial x_2} & \dfrac{\partial^2 K(\mathbf{X},\mathbf{X}')}{\partial x_2 \partial x_1'} & \dfrac{\partial^2 K(\mathbf{X},\mathbf{X}')}{\partial x_2 \partial x_2'} & \cdots & \dfrac{\partial^2 K(\mathbf{X},\mathbf{X}')}{\partial x_2 \partial x_D'} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \dfrac{\partial K(\mathbf{X},\mathbf{X}')}{\partial x_D} & \dfrac{\partial^2 K(\mathbf{X},\mathbf{X}')}{\partial x_D \partial x_1'} & \dfrac{\partial^2 K(\mathbf{X},\mathbf{X}')}{\partial x_D \partial x_2'} & \cdots & \dfrac{\partial^2 K(\mathbf{X},\mathbf{X}')}{\partial x_D \partial x_D'} \end{bmatrix}.$$

Correspondingly, the mean and variance of the posterior predictive distribution of the partial derivative of the energy with respect to coordinate d at $\mathbf{x}^*$ are given as

$$\mathrm{E}\left[\frac{\partial f(\mathbf{x}^*)}{\partial x_d^*}\middle|\mathbf{y}_{\text{ext}},\mathbf{X},\boldsymbol{\theta}\right] = \frac{\partial \mathbf{K}_{\text{ext}}^*}{\partial x_d^*}\left(\mathbf{K}_{\text{ext}} + \sigma^2\mathbf{I}\right)^{-1}\mathbf{y}_{\text{ext}} \tag{3}$$

and

$$\mathrm{Var}\left[\frac{\partial f(\mathbf{x}^*)}{\partial x_d^*}\middle|\mathbf{y}_{\text{ext}},\mathbf{X},\boldsymbol{\theta}\right] = \frac{\partial^2 k(\mathbf{x}^*,\mathbf{x}^{*\prime})}{\partial x_d^*\partial x_d^{*\prime}} - \frac{\partial \mathbf{K}_{\text{ext}}^*}{\partial x_d^*}\left(\mathbf{K}_{\text{ext}} + \sigma^2\mathbf{I}\right)^{-1}\frac{\partial \mathbf{K}_{\text{ext}}^{*}}{\partial x_d^*}^{\mathsf{T}}.$$

Even if the observations are assumed to be accurate, a small but positive value for the noise variance σ^2 is used to avoid numerical problems when inverting the covariance matrix $\mathbf{K}_{\text{ext}}$.

IV. ALGORITHMS

In this section, the two algorithms using the Gaussian process regression (GPR) approach to accelerate CI-NEB calculations, the all-images-evaluated (AIE) algorithm and the one-image-evaluated (OIE) algorithm, are presented in detail.

A. All-images-evaluated (AIE) algorithm

```
Input: A GP model, energy (and zero force) at the two minima on the energy surface, coordinates of the $N_{\text{im}}$ images on the initial path, a final convergence threshold $T_{\text{MEP}}$ for the minimum energy path, an additional final convergence threshold $T_{\text{CI}}$ for the climbing image $i_{\text{CI}}$, a preliminary convergence threshold $T_{\text{CIon}}^{\text{GP}}$ for turning climbing image mode on during the relaxation phase, a maximum displacement $r_{\text{max}}$ of any image from the nearest observed data point.
Output: A minimum energy path represented by $N_{\text{im}}$ images, one of which has climbed to the highest saddle point.
1. Start from the initial path and repeat the following (outer iteration loop):
   A. Evaluate the true energy and force at the $N_{\text{im}} - 2$ intermediate images of the current path and add them to the training data.
   B. Calculate the accurate NEB force vector $F_{\text{NEB}}(i)$ for each intermediate image $i \in \{2, 3, \ldots, N_{im} - 1\}$.
   C. If $\max_i |F_{\text{NEB}}(i)| < T_{\text{MEP}}$ and $|F_{\text{NEB}}(i_{\text{CI}})| < T_{\text{CI}}$, then stop the algorithm (final convergence reached).
   D. Optimize the hyperparameters of the GP model based on the training data and calculate the matrix inversion in Eq. (1).
   E. Relaxation phase: Start from the initial path, set climbing image mode off, and repeat the following (inner iteration loop):
      I. Calculate the GP posterior mean energy and gradient at the intermediate images using Eqs. (1) and (3).
      II. Calculate the approximate NEB force vector $F_{\text{NEB}}^{\text{GP}}(i)$ for each intermediate image using the GP posterior mean gradient.
      III. If climbing image mode is off and $\max_i |F_{\text{NEB}}^{\text{GP}}(i)| < T_{\text{CIon}}^{\text{GP}}$, then turn climbing image mode on and recalculate the approximate NEB force vectors.
      IV. If climbing image mode is on and $\max_i |F_{\text{NEB}}^{\text{GP}}(i)| < \frac{1}{10}T_{\text{CI}}$, then stop the relaxation phase (E).
      V. Move the intermediate images along the approximate NEB force vector $F_{\text{NEB}}^{\text{GP}}(i)$ with a step size defined by the projected velocity Verlet algorithm.
      VI. If the distance from any current image to the nearest observed data point is larger than $r_{\text{max}}$, then reject the last inner step and stop the relaxation phase (E).
```

A pseudocode for the AIE algorithm is presented above. Figure 1 shows an illustration of the progression of the algorithm on the two-dimensional Müller-Brown energy surface.[25] The energy (and the zero gradient of the energy) at the initial and final state minima is assumed to be provided as input.

The algorithm is started by evaluating the energy and force at the $N_{\text{im}} - 2$ intermediate images of an initial path and constructing a GP model for the energy based on the obtained information. The path is then relaxed on the approximate energy surface (GPR iteration 1), which is given as the posterior mean of the GP model, with a regular CI-NEB method using the posterior mean gradient of the GP model to calculate the approximate NEB force at the images. After each relaxation phase, final convergence of the path is checked by evaluating the true energy and force at the images of the relaxed

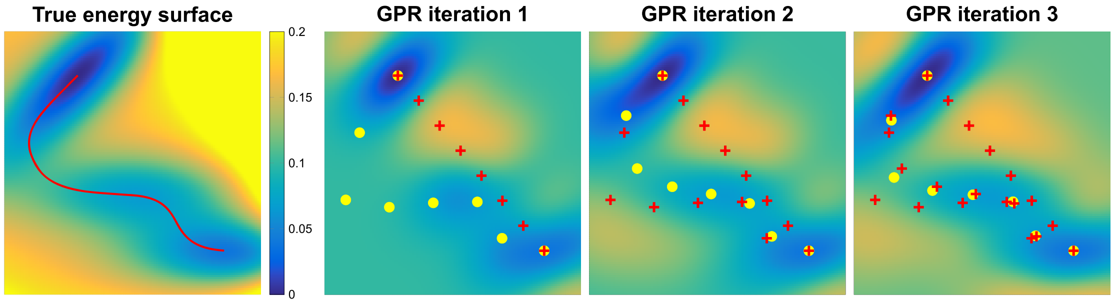


FIG. 1. Far left: The two-dimensional Müller-Brown energy surface, which has three minima, and the minimum energy path (red curve). Three panels to the right: An illustration of the iterative construction of an approximate energy surface in the vicinity of the minimum energy path using the all-images-evaluated algorithm where energy and atomic forces are evaluated at all intermediate images of the nudged elastic band. The initial path is a straight line interpolation between the initial and final state minima. The red + signs show the points at which the energy and atomic forces have been evaluated. The yellow disks show the climbing image nudged elastic band relaxed on the approximate energy surface of each Gaussian process regression iteration. After each GPR iteration, final convergence of the path is checked by energy and force evaluations, which are then added to the training data for the following GPR iteration. After three iterations (and a total of 24 energy and force evaluations), final convergence is confirmed as the magnitude of the NEB force is below the threshold 0.01, both for the climbing image and the other intermediate images.

path, and these observations are then added to the training data to improve the GP model on the following round. As can be seen from Fig. 1, a fairly accurate approximation of the Müller-Brown energy surface is obtained already after three GPR iterations of the AIE algorithm. This corresponds to 18 energy and force evaluations since the path is represented by six movable images in this case. A simplified flowchart of the algorithm is presented in Fig. 2.

The final convergence of the climbing image nudged elastic band is defined by the magnitude of the NEB force (including the spring force parallel to the path tangent) at each image calculated using the energy and force of the true energy surface. Two separate final convergence thresholds are used: T_{MEP} for the maximum NEB force magnitude $\max_i |F_{\mathrm{NEB}}(i)|$ among the intermediate images i and a tighter T_{CI} for the NEB force magnitude $|F_{\mathrm{NEB}}(i_{\mathrm{CI}})|$ of the climbing image i_{CI}.

To ensure that the incomplete relaxation of the path on the approximate energy surface does not disturb final convergence, the convergence threshold $T_{\mathrm{MEP}}^{\mathrm{GP}}$ for the maximum approximate NEB force magnitude $\max_i |F_{\mathrm{NEB}}^{\mathrm{GP}}(i)|$ during the relaxation phase is defined as 1/10 of the tighter final threshold T_{CI}. To decrease the amount of inner iterations during the relaxation phase, there is an alternative option for $T_{\mathrm{MEP}}^{\mathrm{GP}}$ to be defined as 1/10 of the smallest true NEB force magnitude obtained so far on any of the intermediate images (but not less than $T_{\mathrm{CI}}/10$). If the approximation error is assumed not to decrease more than that during one GPR iteration, there is no need for a tighter convergence on the approximated surface and thus the relaxation phase can be sped up by using a larger tolerance. The divisor 10 can also be replaced by some other suitable number.

To make it more certain that the path converges to the same MEP as the one obtained by the regular CI-NEB method, each relaxation phase on the approximate energy surface is started from the same initial path. 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. Starting each relaxation phase from the initial path may possibly improve the stability of the algorithm, but there is also an alternative option which may decrease the number of inner iterations. In this alternative scheme, each relaxation phase would be started from the latest evenly spaced path converged to $T_{\mathrm{CIon}}^{\mathrm{GP}}$, and the climbing image phase would be started from the latest converged CI-NEB path if the climbing image of that path has the highest energy also on the current approximate energy surface. If, instead, the index of the highest energy image has changed, the climbing image phase would be started normally from the preliminarily relaxed evenly spaced path.

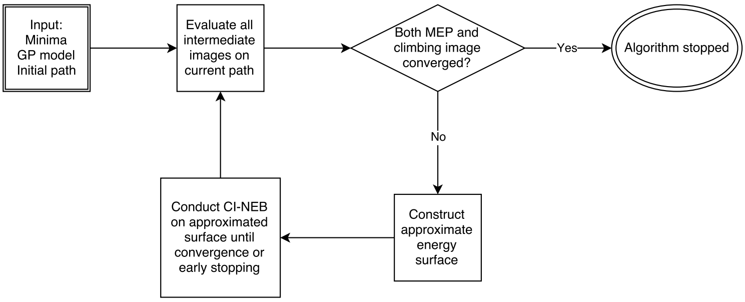

Input:
Minima
GP model
Initial path
Evaluate all intermediate images on current path
Both MEP and climbing image converged?
Yes
Algorithm stopped
No
Construct approximate energy surface
Conduct CI-NEB on approximated surface until convergence or early stopping

FIG. 2. A flowchart of the all-images-evaluated algorithm, where energy and atomic forces are evaluated at all intermediate images of the climbing image nudged elastic band relaxed on the GP-approximated energy surface.

In the early phases of the algorithm, when little information is available about the energy surface, there is a greater possibility that the path wanders far away from the initial path. To prevent this behavior, it is good to have some early stopping rule for the movement of the path. Thus, if the distance from any current image to the nearest observed data point becomes larger than $r_{\max}$, then the last inner step is rejected and the relaxation phase stopped. In the heptamer island benchmark described later in Sec. V, $r_{\max}$ is defined as half of the length of the initial path (sum of the distances between adjacent images), but other definitions, e.g., based on the length scale of the GP model, are also possible.

When the final goal is to estimate the transition rates using harmonic transition state theory, the Hessian matrices at the initial and final state minima will need to be calculated. The Hessian is usually calculated with a finite difference method, where energy and force evaluations are made in the neighborhood of the minima. If these calculations, which anyway are needed for the Hessian, are evaluated already in the beginning of the MEP calculation, the calculated values can be added to the initial data set for the GPR calculations to provide information about the shape of the energy surface around the endpoints of the path and improve especially the early phase of the algorithm. To test the effect of the Hessian input in the heptamer island benchmark, a finite difference displacement of 10^{-3} Å is made in the positive direction along each of the atom coordinates, and the values of the energy and force at these points are included as input in the GPR calculation.

Since the GPR calculations using gradient observations require an inversion of an $(1 + D)N \times (1 + D)N$ matrix, the computational effort scales as $\mathcal{O}(((1 + D)N)^3)$, where N is the number of observation points and D is the number of degrees of freedom (here coordinates of movable atoms). As usual, the matrix inversion is computed by forming a Cholesky decomposition and solving a linear system of equations. Since the model stays the same during the relaxation phase on the approximate energy surface, the matrix inversion needs to be computed only once for each GPR iteration and the complexity of one inner iteration on the approximate energy surface is $\mathcal{O}((1 + D)N)$.

B. One-image-evaluated (OIE) algorithm

Input: A GP model, energy (and zero force) at the two minima on the energy surface, coordinates of the N_{im} images on the initial path, a final convergence threshold T_{MEP} for the minimum energy path, an additional final convergence threshold T_{CI} for the climbing image i_{CI}, a preliminary convergence threshold $T_{\mathrm{CIon}}^{\mathrm{GP}}$ for turning climbing image mode on during the relaxation phase, a maximum displacement $r_{\max}$ of any image from the nearest observed data point.

Output: A minimum energy path represented by N_{im} images, one of which has climbed to the highest saddle point.

```
1. Optimize the hyperparameters of the GP model based on the initial data.
2. Start from the initial path and repeat the following (outer iteration loop):
   A. Calculate the GP posterior variance at the unevaluated images $i \in \mathrm{I_u}$ on the current path using Eq. (2).
   B. Evaluate the true energy and force at the image with highest posterior variance and add them to the training data.
   C. Calculate the accurate NEB force vector $F_{\mathrm{NEB}}(i)$ for the evaluated images $i \in \mathrm{I_e}$.
   D. If all images on the current path have been evaluated, $\max_i |F_{\mathrm{NEB}}(i)| < T_{\mathrm{MEP}}$ and $|F_{\mathrm{NEB}}(i_{\mathrm{CI}})| < T_{\mathrm{CI}}$, then stop the algorithm (final convergence reached).
   E. Reoptimize the GP hyperparameters, calculate the GP posterior mean energy and gradient at the unevaluated images $i \in \mathrm{I_u}$ using Eqs. (1) and (3), and save the matrix inversion for further use.
   F. Calculate the approximate NEB force vector $F_{\mathrm{NEB}}^{\mathrm{GP}}(i)$ for the unevaluated images $i \in \mathrm{I_u}$ using the GP posterior mean gradient and set $F_{\mathrm{NEB}}^{\mathrm{GP}}(i) = F_{\mathrm{NEB}}(i)$ for the evaluated images $i \in \mathrm{I_e}$.
   G. If $\max_i |F_{\mathrm{NEB}}^{\mathrm{GP}}(i)| < T_{\mathrm{MEP}}$:
      I. If $i_{\mathrm{CI}} \in I_u$, then evaluate the energy and force at the climbing image $i_{\mathrm{CI}}$, add them to the training data, and go to C.
      II. If $i_{\mathrm{CI}} \in I_e$ and $|F_{\mathrm{NEB}}(i_{\mathrm{CI}})| < T_{\mathrm{CI}}$, then go to A.
      III. If $i_{\mathrm{CI}} \in I_e$ and $|F_{\mathrm{NEB}}(i_{\mathrm{CI}})| \geq T_{\mathrm{CI}}$, then execute the relaxation phase (H), evaluate the energy and force at the climbing image, add them to the training data, and go to C.
   H. Relaxation phase: Start from the initial path, set climbing image mode off, and repeat the following (inner iteration loop):
      I. Calculate the GP posterior mean energy and gradient at the intermediate images using Eqs. (1) and (3).
      II. Calculate the approximate NEB force vector $F_{\mathrm{NEB}}^{\mathrm{GP}}(i)$ for each intermediate image using the GP posterior mean gradient.
      III. If climbing image mode is off and $\max_i |F_{\mathrm{NEB}}^{\mathrm{GP}}(i)| < T_{\mathrm{CIon}}^{\mathrm{GP}}$, then turn climbing image mode on and recalculate the approximate NEB force vectors.
      IV. If climbing image mode is on and $\max_i |F_{\mathrm{NEB}}^{\mathrm{GP}}(i)| < T_{\mathrm{MEP}}^{\mathrm{GP}} = \frac{1}{10} T_{\mathrm{CI}}$, then stop the relaxation phase (H).
      V. Move the intermediate images along the approximate NEB force vector $F_{\mathrm{NEB}}^{\mathrm{GP}}(i)$ with a step size defined by the projected velocity Verlet algorithm.
      VI. If the distance from any current image $i$ to the nearest observed data point is larger than $r_{\max}$, then reject the last inner step, evaluate image $i$, and go to C.
```

A pseudocode for the OIE algorithm is presented above. Figure 3 shows an illustration of the progression of the algorithm on the Müller-Brown energy surface. Both the GP approximation to the energy surface and the estimated uncertainty after one, two, three, and eleven GPR iterations are shown. The energy (and the zero gradient of the energy) at the initial and final state minima is assumed to be given as input.

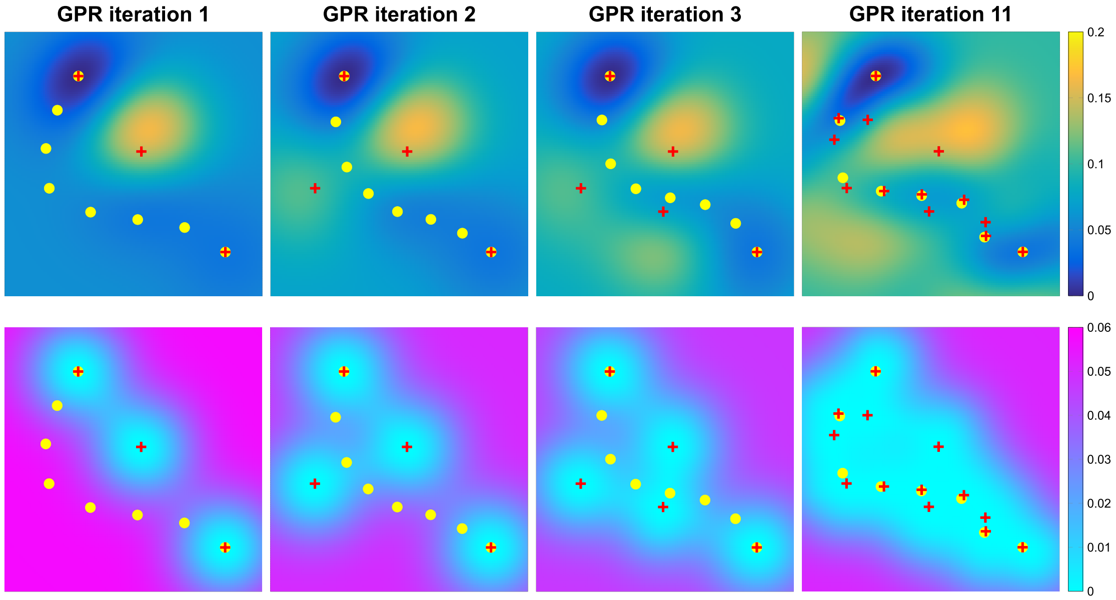


FIG. 3. An illustration of the iterative construction of an approximate energy surface to the two-dimensional Müller-Brown energy surface (shown in Fig. 1) in the vicinity of the minimum energy path using the one-image-evaluated algorithm where the energy and atomic forces are evaluated only at one image of the nudged elastic band. The initial path is a straight line interpolation between the initial and final state minima. Upper panel: The red + signs show the points at which the energy and atomic forces have been evaluated. The yellow disks show the climbing image nudged elastic band relaxed on the approximate energy surface of each Gaussian process regression iteration. After GPR iterations 1, 2, and 3, the energy and force are calculated at the image where the estimated uncertainty is largest and the observed data are then added to the training data set for the following GPR iteration. Lower panel: The estimated uncertainty (standard deviation) of the energy approximation shown directly above. After eleven iterations, the path is not displaced further but the final convergence is checked by evaluating the energy and force at each intermediate image one by one. After 17 evaluations, final convergence is confirmed as the magnitude of the NEB force is below the threshold 0.01 both for the climbing image and the other intermediate images.

The algorithm is started by evaluating the true energy and force at the image located in the middle of the initial path where the uncertainty of the initial GP model is largest. The GP model is then updated based on the obtained information, and the path is relaxed on the approximate energy surface (GPR iteration 1). After each relaxation phase, the true energy and force are evaluated at only one image of the relaxed path before updating the GP model. According to the main rule, the image with the highest uncertainty estimate is selected for evaluation, and the information obtained is then used to improve the GP model on the following round. As can be seen from Fig. 3, a fairly accurate approximation of the Müller-Brown energy surface is obtained already after eleven GPR iterations of the OIE algorithm. This corresponds to only eleven energy and force evaluations, quite a bit fewer than the 18 included in the three GPR iterations of the AIE algorithm shown in Fig. 1.

The details of the OIE algorithm are otherwise similar to the AIE algorithm, but since only one image is evaluated between the GPR iterations, relaxing the path between each evaluation would mean that the accurate NEB force is only known for one image at a time. Thus, it would not be known for sure whether the path has converged on a true MEP. Approximations for the NEB forces can of course be calculated at the unevaluated images based on the updated GP model, but since the NEB forces have been relaxed to zero based on the previous GP approximation and since the largest changes to the approximation usually emerge near the new observation point, it is most likely that these approximations underestimate the NEB force magnitudes. The approximated NEB forces, however, together with the accurate ones, at least indicate if there is a possibility that the path may have converged. The general idea for the convergence check of the OIE algorithm is that when the maximum magnitude of both the accurate and approximated NEB forces is below the final convergence threshold T_{MEP}, more images are evaluated without moving the path until some of the magnitudes rise above the threshold or all images have been evaluated. Since the images with the highest uncertainty, which are the most likely ones to violate the convergence criterion, are evaluated first, it is likely that the convergence check will be interrupted early if the path has not yet truly converged.

The special role of the climbing image makes the evaluation rules a bit more complicated. Since the climbing image has a tighter convergence threshold and since the position of the climbing image also affects the distribution of the other images, it is desirable to favor evaluations of the climbing image during the convergence check. As long as the maximum magnitude of the accurate and approximated NEB forces is above T_{MEP}, the GP relaxation phase is executed normally and the image with the highest uncertainty is evaluated. After the maximum magnitude has reduced below T_{MEP}, the

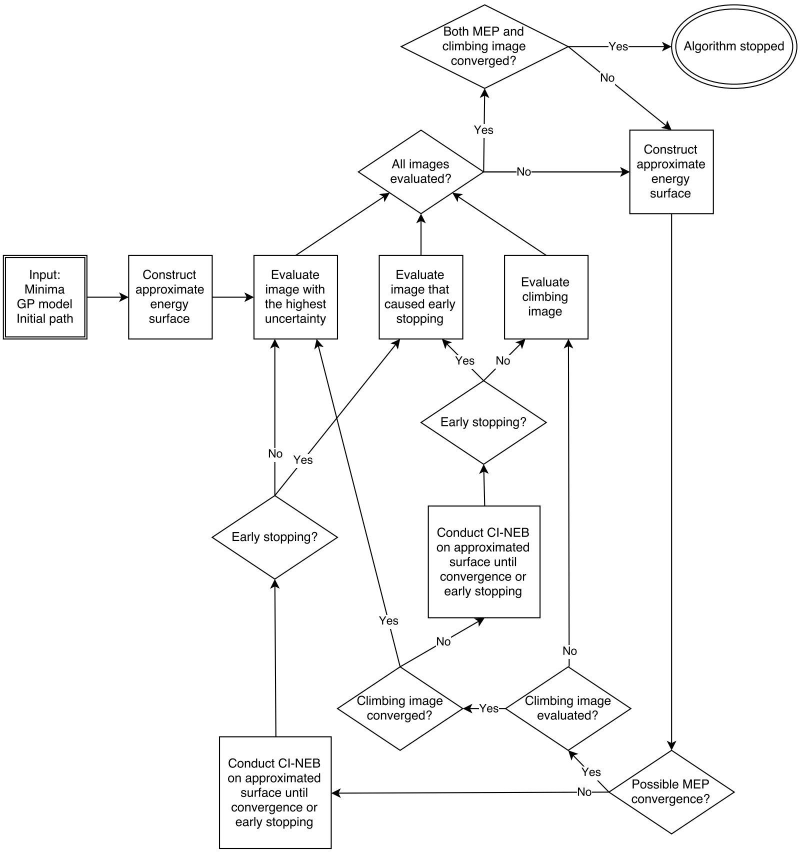

Both MEP and climbing image converged?
Yes
Algorithm stopped
No
Yes
All images evaluated?
No
Construct approximate energy surface
Input: Minima GP model Initial path
Construct approximate energy surface
Evaluate image with the highest uncertainty
Evaluate image that caused early stopping
Evaluate climbing image
Yes
No
Early stopping?
No
Yes
Early stopping?
Conduct CI-NEB on approximated surface until convergence or early stopping
Yes
No
No
Climbing image converged?
Yes
Climbing image evaluated?
Yes
No
Possible MEP convergence?
Conduct CI-NEB on approximated surface until convergence or early stopping

FIG. 4. A flowchart of the one-image-evaluated algorithm, where the energy and atomic forces are evaluated at only one image before a new Gaussian process regression iteration.

climbing image is evaluated without moving the path (if not already evaluated). As long as the maximum NEB force magnitude stays below T_{MEP} but the NEB force magnitude $|F_{\mathrm{NEB}}(i_{\mathrm{CI}})|$ on the climbing image is above T_{CI}, the path is relaxed and the climbing image is re-evaluated. Finally, if the maximum magnitude of the accurate and approximated NEB forces is below T_{MEP}, the climbing image has been evaluated and $|F_{\mathrm{NEB}}(i_{\mathrm{CI}})| < T_{\mathrm{CI}}$, then more images are evaluated without moving the path, starting from the image with the highest uncertainty.

Another exception to the selection of the image to be evaluated is caused by the early stopping rule during the relaxation phase. If the distance from any current image i to the nearest observed data point becomes larger than $r_{\max}$, then the last inner step is rejected, the relaxation phase stopped, and image i evaluated next. A simplified flowchart of the OIE algorithm is presented in Fig. 4.

V. APPLICATION TO THE HEPTAMER ISLAND BENCHMARK

A test problem that has been used in several studies of algorithms for finding MEPs and saddle points involves an island of seven atoms on the (111) surface of a face-centered cubic (FCC) crystal.[34,35] The interaction between the atoms is described with a simple Morse potential to make the implementation of the benchmark easy. The initial, saddle point and final configurations of the atoms for the 13 lowest activation energy transitions, labeled from A to M, are shown in Fig. 5. In the initial state, the seven atoms sit at FCC surface sites and form a compact island. In transitions A and B, the whole island shifts to 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 a pair of edge atoms moves in such a way that one of the atoms is displaced away from the island while the other atom takes its place.

The calculations were carried out using five intermediate images ($N_{\mathrm{im}} = 7$) in the CI-NEB calculations starting with an IDPP path, and the images were moved iteratively to an MEP using the projected velocity Verlet algorithm[14] with a time step of 0.1 fs. A time step of 1 fs is too large and leads to overshooting, but a time step of 0.01 fs requires a significantly larger number of iterations. The algorithms were continued until the magnitude of the true NEB force acting on the climbing image had dropped below $T_{\mathrm{CI}} = 0.01$ eV/Å. A larger tolerance, T_{MEP} = 0.3 eV/Å, was used for the magnitude of the NEB force acting on the other images in the CI-NEB calculation. During each relaxation phase on the GP-approximated surface, the

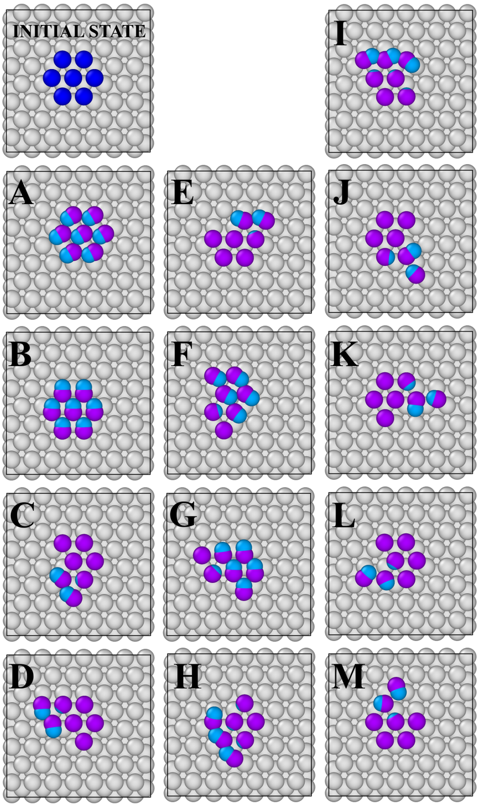


FIG. 5. An illustration of the atomic transitions of the heptamer island benchmark problem. The initial configuration involves a seven-atom island (dark blue disks in the uppermost left column) adsorbed at FCC sites of an FCC(111) surface. Saddle point configuration (light blue disks) and final configuration (purple disks) are shown together for each of the transitions, labeled *A–M*.

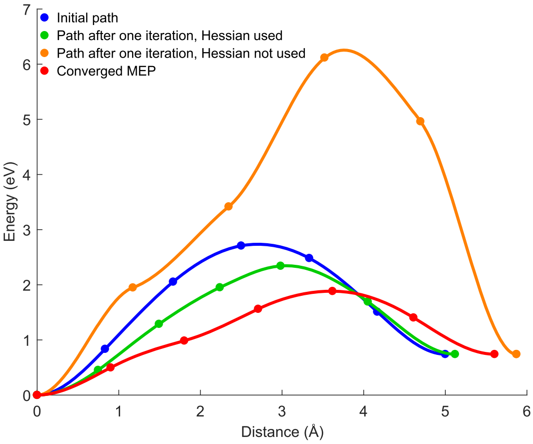


FIG. 6. An illustration of the improvement that can be obtained by using the Hessian at the initial and final state minima. The calculations are for transition *I* when the substrate atoms are frozen. The green and orange points show the true energy evaluated at the location of images of a climbing image nudged elastic band relaxed on the first approximate energy surface of the all-images-evaluated algorithm, i.e., at this point, the energy and force had been evaluated at all the images of the initial path. The path obtained when the Hessian is not used (orange) is far from the converged minimum energy path (red), quite a bit farther away than the initial path (blue), and the climbing image has moved to the final state minimum. The path obtained when the Hessian is used (green), however, represents a clear improvement to the initial path, having moved significantly closer to the converged MEP. Eventually, the path converges after 14 Gaussian process regression iterations (a total of 75 energy and force evaluations required to confirm final convergence) when the Hessian is used, and after 17 GPR iterations when it is not used.

preliminary convergence threshold $T^{\mathrm{GP}}_{\mathrm{CIon}}$ for turning climbing image mode on was 1 eV/Å. The GPR calculations were carried out using the GPstuff toolbox.[36] The fixed parameters of the GP model were chosen to be $\sigma^2 = 10^{-8}$ eV^2 and $\sigma_c^2 = 100$ eV^2. A common length scale $l_d = l$ was used for all dimensions $d = 1, \ldots, D$, and a zero mean Student's t-distribution (restricted to positive values) with 1 Å^2 scale and four degrees of freedom was used as a prior distribution for l and a log-uniform distribution for σ_m^2 (i.e., the default priors of GPstuff) in the optimization of the hyperparameters. The prior distributions stabilize the point estimates of the hyperparameters especially in the beginning, when there is little data available. The optimization was performed using the scaled conjugate gradient algorithm.[37]

By using input from the Hessian at the initial and final state minima, the path relaxed on the GP-approximated energy surface can become qualitatively similar to the true MEP with fewer GPR iterations. This is illustrated for transition *I* in Fig. 6. It shows the true energy evaluated at the location of the images of the initial path, the converged MEP, and the path after one GPR iteration in the AIE algorithm with and without input from the Hessian. The estimate without the Hessian input is quite poor at this point, the path reaching an area of high energy and the climbing image moving to the final state minimum.

The reason for this behavior can be seen from Fig. 7, where the GP-approximated energy at the location of the images after one GPR iteration (five energy and force evaluations) is shown. The maximum of the approximated energy along the path is indeed at the final state of the path. However, with the input from the Hessian, a qualitatively correct path is obtained already after one GPR iteration. In this case, the information coming from the Hessian about the curvature at the endpoints ensures that the GP-approximated surface has minima at those points. Without the Hessian input, the three subsequent GPR iterations still show qualitatively wrong variation of the energy along the path, and it is only after the fifth iteration that enough input has been obtained for the GP model to be reasonably accurate to produce a path qualitatively similar to the true MEP.

The overall reduction in the number of energy and force evaluations corresponds essentially to the first three GPR iterations that can be skipped when the Hessian is provided. It takes 17 GPR iterations without and 14 iterations with the Hessian input to reach convergence in this case. The energy and force evaluations needed to construct the Hessian are not counted in the cost of finding the MEP since they need to be carried out anyway if the transition rate is estimated using harmonic transition state theory. While the reduction in the number of

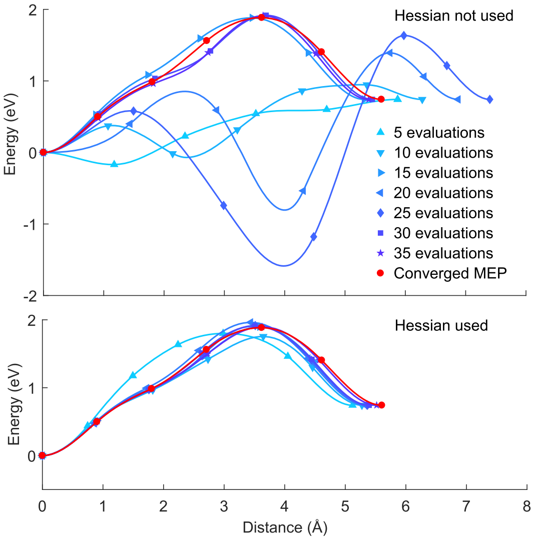


FIG. 7. Comparison of the performance of the all-images-evaluated algorithm with and without input from the Hessian at the initial state minima. The markers show the GP-approximated energy at the images of the climbing image nudged elastic band relaxed on the approximate energy surface after one to seven Gaussian process regression iterations (5 to 35 energy and force evaluations, blue) and after final convergence (red) for transition *I* when the substrate atoms are frozen. Without the Hessian input (upper graph), the approximate energy surface after one GPR iteration does not have an intermediate barrier and the climbing image moves to the final state minimum. The true energy evaluated at each image along this path is shown in Fig. 6 and shows a large energy barrier. It takes six GPR iterations (30 energy and force evaluations) before the path relaxed on the approximated surface starts to look qualitatively similar to the converged minimum energy path. With the Hessian input (lower graph), the energy along the path relaxed on the approximate energy surface is qualitatively similar to that of the converged MEP already after one iteration.

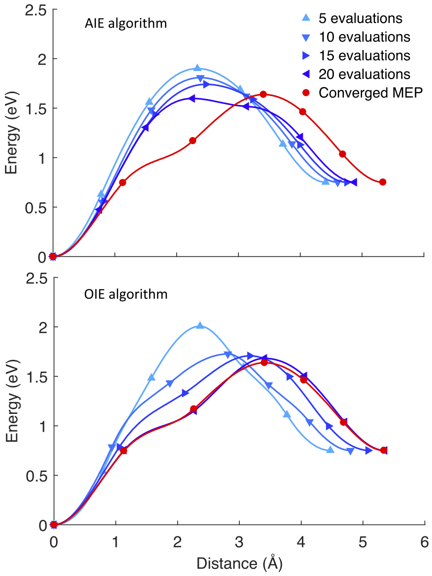


FIG. 8. Comparison of the performance of the all-images-evaluated (AIE) and the one-image-evaluated (OIE) algorithms (Hessian input used) for transition *F* when the substrate atoms are frozen. The markers show the GP-approximated energy at the images of the climbing image nudged elastic band relaxed on the approximate surface after 5, 10, 15, and 20 energy and force evaluations (blue) and after final convergence (red). With the AIE algorithm (upper figure), the variation of the energy is significantly different for the path on the approximate surface compared with the true minimum energy path even after 20 evaluations (four Gaussian process regression iterations). With the OIE algorithm (lower figure), the variation of the energy along the path on the approximate surface is close to that of the converged MEP after 20 evaluations (corresponding to 20 GPR iterations in this case).

energy and force evaluations (here 15) corresponds to savings of about 20%, the importance of the Hessian input can be greater in more challenging systems when avoiding exploration of regions far away from the MEP where the atomic forces tend to be large.

Even larger savings are obtained by using the uncertainty estimate provided by the GP model to evaluate the energy and force only at the image where the true energy is the most poorly known instead of all images, i.e., to move to the OIE from the AIE algorithm. This is illustrated for transition *F* in Fig. 8, where the GP-approximated energy at the CI-NEB images is shown after a certain number of energy and force evaluations for the AIE and OIE algorithms. In the AIE case, the path is still qualitatively incorrect after 20 evaluations (four GPR iterations), while this suffices for the OIE algorithm (20 GPR iterations) to nearly reach convergence. For the AIE algorithm, it takes a total of 75 energy and force evaluations to confirm final convergence, while the OIE algorithm requires 39. Similar reduction in the number of energy and force evaluations was found for the other transitions.

The number of energy and force evaluations required to confirm final convergence for each of the 13 transitions using the AIE and OIE algorithms is given in Table I as a fraction of the number of evaluations required by a regular CI-NEB method. Also, the effect of using the Hessian input for the AIE algorithm is shown. These numbers correspond to the case where six nearest substrate atoms can move in addition to the seven island atoms. The relative number of evaluations compared to the regular CI-NEB varies between the transitions. A clear trend is that the more complex the transition and the more the iterations required by the regular CI-NEB, the larger is the relative effect of the GPR approach.

The average number of energy and force evaluations for transitions *C*–*M* as a function of the number of degrees of freedom is shown in Fig. 9.[38] For the smallest number of degrees of freedom, 21, only the seven island atoms are allowed to move while all the substrate atoms are frozen. For the larger numbers of degrees of freedom, some of the surface atoms are also allowed to move. Starting with the AIE algorithm, the use of the Hessian input reduces the number of evaluations by about 20%, but the transition to the OIE algorithm has an even larger effect, a reduction to a half.

The OIE results represent savings of an order of magnitude with respect to the regular CI-NEB calculation. The number of energy and force evaluations reported here for the CI-NEB method is similar to what has been reported earlier for this test problem.[34,35] It is possible to use a more efficient minimization

TABLE I. The number of energy and force evaluations needed to converge the regular climbing image nudged elastic band (CI-NEB) calculations of the heptamer island benchmark transitions (shown in Fig. 5) when 39 degrees of freedom are included, and the reduction in the number of evaluations obtained with the Gaussian process regression approach using the all-images-evaluated algorithm without the Hessian input (AIE), all-images-evaluated algorithm with the Hessian input (AIE-H), and one-image-evaluated algorithm without the Hessian input (OIE).

	Number of evaluations	Number of evaluations as a fraction of evaluations needed for CI-NEB		
Transition	CI-NEB	AIE	AIE-H	OIE
A	120	0.42	0.42	0.13
B	120	0.42	0.42	0.23
C	285	0.25	0.21	0.13
D	265	0.26	0.23	0.14
E	290	0.24	0.19	0.13
F	855	0.12	0.12	0.05
G	840	0.13	0.11	0.05
H	1480	0.08	0.07	0.04
I	1480	0.07	0.07	0.04
J	605	0.15	0.12	0.07
K	610	0.14	0.12	0.07
L	565	0.17	0.12	0.06
M	570	0.17	0.11	0.06

scheme to relax the images in CI-NEB calculations,[28] but the difference is not so large. The test results presented here, therefore, show that the use of GPR can significantly reduce the computational effort in, for example, calculations of MEPs for surface processes.

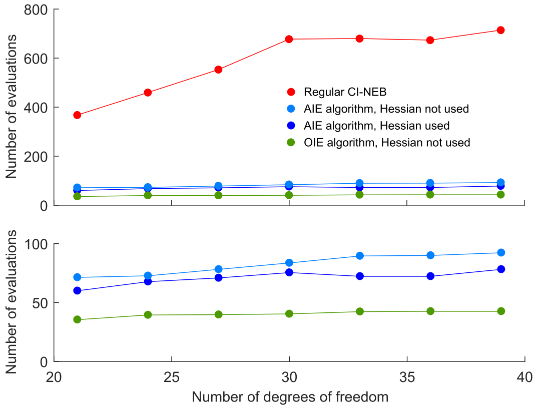


FIG. 9. Average number of energy and force evaluations needed to converge to a minimum energy path in climbing image nudged elastic band calculations of the heptamer island benchmark as a function of the number of degrees of freedom, increased by allowing a larger number of substrate atoms to move. The lower figure shows the same results as the upper figure on a different scale to better distinguish between the various implementations of the Gaussian process regression approach. For the larger numbers of degrees of freedom, the one-image-evaluated (OIE) algorithm provides about 1/20 reduction in the number of energy and force evaluations as compared with a regular CI-NEB calculation. The all-images-evaluated (AIE) algorithm requires about twice as many evaluations as the OIE algorithm, but the use of the Hessian at the initial and final state minima can reduce that by about 20%. The use of the Hessian has less effect when the OIE algorithm is used (not shown).

VI. DISCUSSION

The results presented here show that the GPR approach can reduce the number of energy and force evaluations needed in CI-NEB calculations of MEPs by an order of magnitude. This is important since a large amount of computer time is used in such calculations, especially when *ab initio* or density functional theory calculations are used. Compared with the previous proof-of-principle calculations,[21] three major improvements to the algorithm have been presented here. First of all, the CI-NEB algorithm was used where one of the images is pushed up to the maximum along the MEP. This provides stability and accelerates convergence because it focuses more evaluations of the energy and force in the vicinity of the first-order saddle point, the most important part of the energy surface.

Second, the benefit of using the finite difference estimate of the Hessian at the endpoints was demonstrated and found to result in a 20% reduction in the number of energy and function evaluations needed to converge on an MEP with the AIE algorithm. This estimation of the Hessian does not represent any additional energy and force evaluations in cases where the goal is to calculate the transition rates using harmonic transition state theory. The usual ordering of the calculations is simply changed; the Hessian at the endpoint minima is evaluated before the MEP calculation rather than afterwards. If a higher level of rate theory, such as optimal hyperplanar transition state theory,[6] is used, then analogous information about the initial and final states can be obtained from dynamical trajectories.

While the Hessian input reduced the number of energy and function evaluations only by 20%, a significant advantage is likely in greater stability and lower probability of the path escaping into irrelevant regions of the energy surface in the first few GPR iterations. This we have seen in preliminary studies of dissociative adsorption of H_2 on a metal surface and water molecule diffusion on an ice Ih(0001) surface.[39] The addition of the finite difference data points for the Hessian adds significantly to the memory requirements of the GPR calculation. Those data points could, however, be dropped after just a few GPR iterations since they are not needed when the GP approximation of the energy surface becomes reasonably accurate around the MEP. A more elegant and efficient way of incorporating the information from the Hessian into the covariance calculation could also be developed. The implementation described here represents just an initial test to see how important such information can be.

As a third improvement, a significant reduction in the computational effort was shown to be possible by using the one-image-evaluated algorithm instead of the earlier all-images-evaluated approach. The number of energy and force evaluations was reduced to a half in the heptamer benchmark. In the OIE algorithm, the true energy and force are evaluated only at one image, rather than all images along the path, before a new GPR iteration. A calculated choice of the location of each evaluation can be made based on the uncertainty estimate provided by the GP model. Interestingly, the use of the Hessian did not provide significant reduction in the number of energy and force evaluations required by the OIE algorithm.

This apparently stems from the fact that the OIE algorithm involves fewer evaluations in the early phase of the iterative GPR process where the approximation to the energy surface is poor.

The heptamer island benchmark studied here is a relatively simple example, and it will be important to test the GPR approach on more complex systems to be able to fully assess its utility and to develop the methodology further. On complex energy surfaces, there may exist multiple MEPs connecting the two endpoint minima, which would require some kind of sampling of MEPs.[40] Also, some systems may have multiple local minima and highly curved MEPs, which can lead to convergence problems unless a large number of images are included in the calculation. In systems where various types of molecular interactions are involved, the optimal length scale may vary depending on the location in the coordinate space. In such cases, it may be advantageous to use a GP model that allows different length scales in different parts of the space.

In order to tackle large systems, the scaling of the GPR calculations will need to be improved. A more efficient implementation could be obtained, e.g., by using a compactly supported covariance function to produce a sparse covariance matrix where data points far away from each other become independent.[41] It may also be possible to reduce the dimensionality by using partially additive models, where the interaction term in the energy function for far away atoms is ignored. There is, however, a large set of important problems, such as calculations of catalytic processes often involving rather small molecules adsorbed on surfaces, where the complexity is comparable to the heptamer island benchmark and where the GPR approach is clearly going to offer a significant reduction in computational effort in NEB calculations of MEPs.

At a low enough temperature, quantum mechanical tunneling becomes the dominant transition mechanism, and the task is then to find a minimal action path.[9,42,43] The effect of the GPR approach in tunneling path calculations could be even larger than for MEP calculations since each iteration involves more energy and force calculations (Feynman paths rather than points in configuration space) and thereby more data for the modeling. In addition to atomic rearrangements, it will be valuable to apply the GPR approach also to magnetic transitions where the magnetic properties of the system are evaluated by computationally intensive *ab initio* or density functional theory calculations. The relevant degrees of freedom in magnetic transitions are the angles defining the orientation of the magnetic vectors, and the task is again to find MEPs on the energy surface with respect to those angles.[11–13]

ACKNOWLEDGMENTS

We thank Professor Andrew Peterson for helpful discussions. This work was supported by the Icelandic Research Fund and the Academy of Finland (Grant No. 278260). We acknowledge the computational resources provided by the Aalto Science-IT project and the Computer Services of the University of Iceland.

[1] E. Wigner, Trans. Faraday Soc. **34**, 29 (1938).
[2] H. A. Kramers, Physica **7**, 284 (1940).
[3] J. C. Keck, “Variational theory of reaction rates,” in *Advance in Chemical Physics*, edited by I. Prigogine (John Wiley & Sons, 1967), Vol. 13, pp. 85–121.
[4] G. K. Schenter, G. Mills, and H. Jónsson, J. Chem. Phys. **101**, 8964 (1994).
[5] G. Mills, H. Jónsson, and G. K. Schenter, Surf. Sci. **324**, 305 (1995).
[6] G. H. Jóhannesson and H. Jónsson, J. Chem. Phys. **115**, 9644 (2001).
[7] T. Bligaard and H. Jónsson, Comput. Phys. Commun. **169**, 284 (2005).
[8] G. H. Vineyard, J. Phys. Chem. Solids **3**, 121 (1957).
[9] H. Jónsson, Proc. Natl. Acad. Sci. U. S. A. **108**, 944 (2011).
[10] P. F. Bessarab, V. M. Uzdin, and H. Jónsson, Phys. Rev. B **85**, 184409 (2012).
[11] P. F. Bessarab, V. M. Uzdin, and H. Jónsson, Z. Phys. Chem. **227**, 1543 (2013).
[12] P. F. Bessarab, V. M. Uzdin, and H. Jónsson, Phys. Rev. B **89**, 214424 (2014).
[13] P. F. Bessarab, A. Skorodumov, V. M. Uzdin, and H. Jónsson, Nanosyst.: Phys., Chem., Math. **5**, 757 (2014).
[14] H. Jónsson, G. Mills, and K. W. Jacobsen, “Nudged elastic band method for finding minimum energy paths of transitions,” in *Classical and Quantum Dynamics in Condensed Phase Simulations*, edited by B. J. Berne, G. Ciccotti, and D. F. Coker (World Scientific, 1998), pp. 385–404.
[15] P. F. Bessarab, V. M. Uzdin, and H. Jónsson, Comput. Phys. Commun. **196**, 335 (2015).
[16] A. A. Peterson, J. Chem. Phys. **145**, 074106 (2016).
[17] A. O’Hagan and J. F. C. Kingman, J. R. Stat. Soc. B **40**, 1 (1978).
[18] D. J. C. MacKay, “Introduction to Gaussian processes,” in *Neural Networks and Machine Learning*, edited by C. M. Bishop (Springer Verlag, 1998), pp. 133–166.
[19] R. M. Neal, “Regression and classification using Gaussian process priors (with discussion),” in *Bayesian Statistics*, edited by J. M. Bernardo, J. O. Berger, A. P. Dawid, and A. F. M. Smith (Oxford University Press, 1999), Vol. 6, pp. 475–501.
[20] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning* (MIT Press, 2006).
[21] O.-P. Koistinen, E. Maras, A. Vehtari, and H. Jónsson, Nanosyst.: Phys., Chem., Math. **7**, 925 (2016); a slightly corrected version is available as e-print arXiv:1703.10423.
[22] J. Lampinen and A. Vehtari, Neural Networks **14**, 257 (2001).
[23] G. Henkelman, B. P. Uberuaga, and H. Jónsson, J. Chem. Phys. **113**, 9901 (2000).
[24] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. de Freitas, Proc. IEEE **104**, 148 (2016).
[25] K. Müller and L. D. Brown, Theor. Chim. Acta **53**, 75 (1979).
[26] S. Smidstrup, A. Pedersen, K. Stokbro, and H. Jónsson, J. Chem. Phys. **140**, 214106 (2014).
[27] G. Henkelman and H. Jónsson, J. Chem. Phys. **113**, 9978 (2000).
[28] D. Sheppard, R. Terrell, and G. Henkelman, J. Chem. Phys. **128**, 134106 (2008).
[29] A. O’Hagan, “Some Bayesian numerical analysis,” in *Bayesian Statistics*, edited by J. M. Bernardo, J. O. Berger, A. P. Dawid, and A. F. M. Smith (Oxford University Press, 1992), Vol. 4, pp. 345–363.
[30] C. E. Rasmussen, “Gaussian processes to speed up hybrid Monte Carlo for expensive Bayesian integrals,” in *Bayesian Statistics*, edited by J. M. Bernardo, A. P. Dawid, J. O. Berger, M. West, D. Heckerman, M. J. Bayarri, and A. F. M. Smith (Oxford University Press, 2003), Vol. 7, pp. 651–659.
[31] E. Solak, R. Murray-Smith, W. E. Leithead, D. J. Leith, and C. E. Rasmussen, “Derivative observations in Gaussian process models of dynamic systems,” in *Advances in Neural Information Processing Systems*, edited by S. Becker, S. Thrun, and K. Obermayer (MIT Press, 2003), Vol. 15, pp. 1057–1064.
[32] J. Riihimäki and A. Vehtari, “Gaussian processes with monotonicity information,” in *Proceedings of Machine Learning Research*, edited by Y. W. Teh and M. Titterington (PMLR, 2010), Vol. 9, pp. 645–652.
[33] A. P. Bartók and G. Csányi, Int. J. Quantum Chem. **115**, 1051 (2015).
[34] G. Henkelman, G. H. Jóhannesson, and H. Jónsson, “Methods for finding saddle points and minimum energy paths,” in *Progress in Theoretical Chemistry and Physics*, edited by S. D. Schwartz (Kluwer Academic, 2000), Vol. 5, pp. 269–300.
[35] S. T. Chill, J. Stevenson, V. Ruhle, C. Shang, P. Xiao, J. D. Farrell, D. J. Wales, and G. Henkelman, J. Chem. Theory Comput. **10**, 5476 (2014).
[36] J. Vanhatalo, J. Riihimäki, J. Hartikainen, P. Jylänki, V. Tolvanen, and A. Vehtari, J. Mach. Learn. Res. **14**, 1175 (2013).

[37] C. M. Bishop, "Scaled conjugate gradients," in *Neural Networks for Pattern Recognition* (Clarendon Press, 1995), pp. 282–285.

[38] Transitions *A* and *B* are not included in the averages shown in Fig. 9 because the regular CI-NEB required an anomalously large number of iterations for some of the intermediate numbers of degrees of freedom. The results of the GPR algorithms were, however, similar for all numbers of degrees of freedom tested here.

[39] E. R. Batista and H. Jónsson, Comput. Mater. Sci. **20**, 325 (2001).

[40] E. Maras, O. Trushin, A. Stukowski, T. Ala-Nissilä, and H. Jónsson, Comput. Phys. Commun. **205**, 13 (2016).

[41] J. Vanhatalo and A. Vehtari, "Speeding up the binary Gaussian process classification," in *Proceedings of the 26th Conference on Uncertainty in Artificial Intelligence (UAI, 2010)*, edited by P. Gründwald and P. Spirtes (AUAI Press, 2010), pp. 623–632.

[42] G. Mills, G. K. Schenter, D. E. Makarov, and H. Jónsson, Chem. Phys. Lett. **278**, 91 (1997).

[43] G. Mills, G. K. Schenter, D. E. Makarov, and H. Jónsson, "RAW quantum transition state theory," in *Classical and Quantum Dynamics in Condensed Phase Simulations*, edited by B. J. Berne, G. Ciccotti, and D. F. Coker (World Scientific, 1998), pp. 405–421.

Publication III

Olli-Pekka Koistinen, Vilhjálmur Ásgeirsson, Aki Vehtari, and Hannes Jónsson. Nudged elastic band calculations accelerated with Gaussian process regression based on inverse interatomic distances. *Journal of Chemical Theory and Computation*, volume 15, issue 12, pages 6738–6751, October 2019.

JCTC Journal of Chemical Theory and Computation

Article

Cite This: J. Chem. Theory Comput. 2019, 15, 6738−6751

pubs.acs.org/JCTC

Nudged Elastic Band Calculations Accelerated with Gaussian Process Regression Based on Inverse Interatomic Distances

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

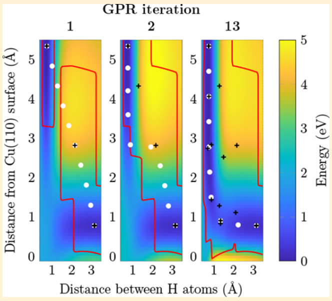
[†]Department of Computer Science, Aalto University, 02150 Espoo, Finland

[‡]Science Institute and Faculty of Physical Sciences, University of Iceland, 107 Reykjavík, Iceland

[§]Department of Applied Physics, Aalto University, 02150 Espoo, Finland

Supporting Information

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 (MEPs) 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 nonparametric 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

Received: July 10, 2019
Published: October 22, 2019

DOI: 10.1021/acs.jctc.9b00692

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 13 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 $3N_m$-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:

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

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 $\mathbf{f} = [f(\mathbf{x}^{(1)}), f(\mathbf{x}^{(2)}), ..., f(\mathbf{x}^{(N)})]^{\mathrm{T}}$ at any finite set of input points $\mathbf{X} = [\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, ..., \mathbf{x}^{(N)}]^{\mathrm{T}}$ is a multivariate Gaussian distribution $p(\mathbf{f}) = \mathcal{N}(\mathbf{m}, K(\mathbf{X}, \mathbf{X}))$, where $\mathbf{m} = [m(\mathbf{x}^{(1)}), m(\mathbf{x}^{(2)}), ..., m(\mathbf{x}^{(N)})]^{\mathrm{T}}$ 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

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

is a $3N_{\mathrm{m}}$-dimensional configuration vector including the Cartesian coordinates for moving atoms 1, 2, ..., $N_{\mathrm{m}} \in A_{\mathrm{m}}$. 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 nonstationary 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:

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

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

DOI: 10.1021/acs.jctc.9b00692
J. Chem. Theory Comput. 2019, 15, 6738−6751

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:

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

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 $\mathbf{y} = [y^{(1)}, y^{(2)}, ..., y^{(N)}]^{\mathrm{T}}$ 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

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

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_{d}^2, the training covariance matrix $K(\mathbf{X}, \mathbf{X})$ is extended correspondingly to include prior covariances between the partial derivatives and function values

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

and the covariances between the derivatives

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

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_{d}^2 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 $p(\boldsymbol{\theta} \mid \mathbf{y}, \mathbf{X}) \propto p(\boldsymbol{\theta})p(\mathbf{y} \mid \mathbf{X}, \boldsymbol{\theta})$, where

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

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_{c}^2 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 $\sigma_d^2 = 10^{-8}$ 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 $T_{\text{MEP}}^{\text{GP}} = T_{\text{CI}}/10$, 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 $T_{\text{CIon}}^{\text{GP}}$ 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 $\mathbf{x}_{\text{im}}$ of the current path, there needs to exist an evaluated configuration $\mathbf{x}_{\text{eval}}$ so that

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

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 $\mathbf{x}_{\text{im}}$ of the current path, there needs to exist an evaluated configuration $\mathbf{x}_{\text{eval}}$ so that

$$\begin{gathered}\forall\ i \in \text{A}_{\text{m}}\ \forall\ j \in \text{A}_{\text{m}} \cup \text{A}_{\text{f}}: \\ \frac{2}{3} r_{i,j}(\mathbf{x}_{\text{eval}}) < r_{i,j}(\mathbf{x}_{\text{im}}) < \frac{3}{2} r_{i,j}(\mathbf{x}_{\text{eval}})\end{gathered} \tag{14}$$

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

DOI: 10.1021/acs.jctc.9b00692

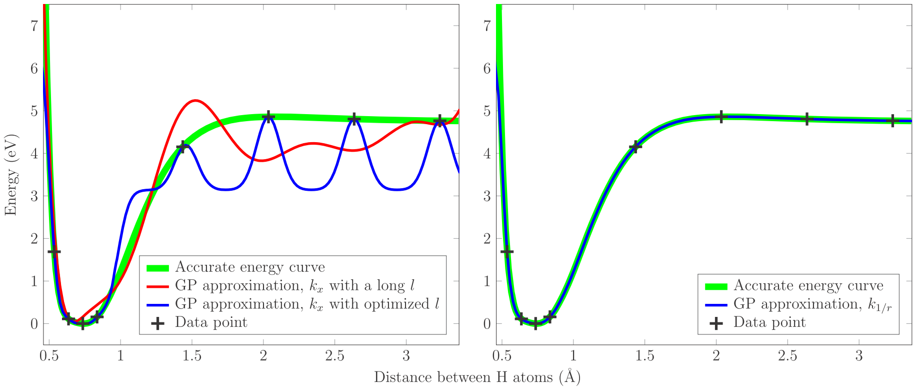


Figure 1. (thick green curve) "True" energy as a function of distance between two hydrogen atoms. Training data for the GP models, marked with + signs, include accurate values for both energy and its first derivative with respect to the coordinate of the moving hydrogen atom. (left) GP approximations 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: $\sigma_m = 1.6$ eV, $l = 1$ Å), and the blue curve shows that with optimized hyperparameters ($\sigma_m \approx 1.9$ 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.

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 $\mathbf{x}_{im}$ is defined as follows: An individual atom $i \in A_m$ cannot move more than 99% of

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

where the minimum is taken over all interatomic distances from that atom to any other atom in $\mathbf{x}_{im}$. 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

DOI: 10.1021/acs.jctc.9b00692

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 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.

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_{CI} = 0.01 eV/Å for the climbing image and below T_{MEP} = 0.3 eV/Å for the other intermediate images. A spring constant of 1 eV/Å^2 was used for all image intervals.

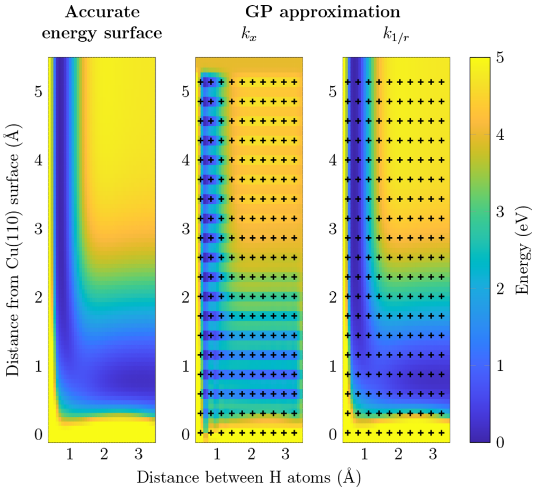


Figure 2. Two-dimensional cut through the potential energy surface for a pair of hydrogen atoms near a Cu(110) surface. The H−H molecular axis is parallel to the surface and perpendicular to the atom rows on the Cu(110) surface. The horizontal axis represents the distance between the two H atoms, and 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.

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 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 13 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

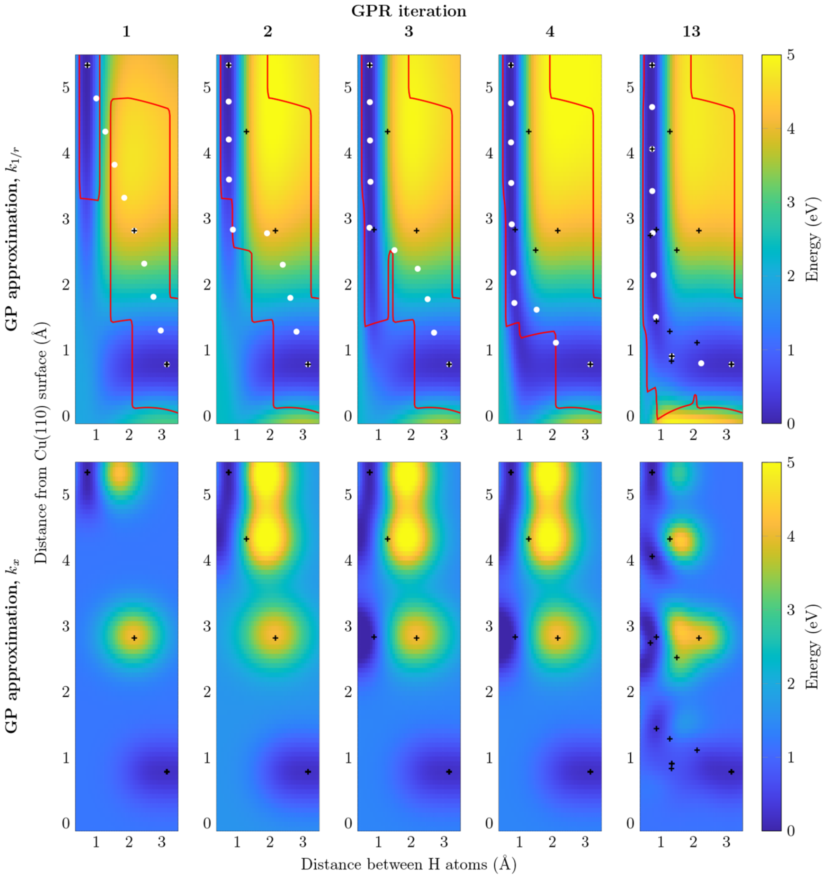


Figure 3. Two-dimensional cut through the potential energy surface for an H_2 molecule dissociating on a Cu(110) surface. The H−H molecular axis is parallel to the surface and perpendicular to the atom rows on the Cu(110) surface. The horizontal axis represents the distance between the two H atoms, and the vertical axis represents the distance between the H atoms and the Cu(110) surface. (upper) GP approximations with covariance function $k_{1/r}$ after 1, 2, 3, 4, and 13 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 13 GPR iterations. (lower) 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.

$\mathcal{D}_x$ 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

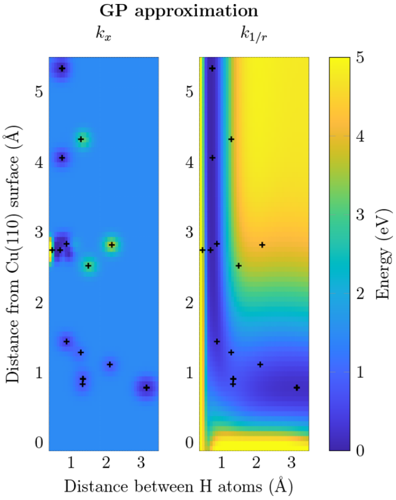


Figure 4. Illustrations of GP approximations based on covariance functions k_x (left) and $k_{1/r}$ (right), corresponding to the rightmost graphs in Figure 3 after adding one high-gradient training data point near the left border of the graph and reoptimizing the hyperparameters.

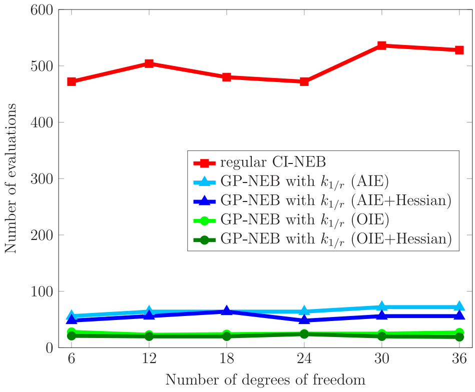


Figure 5. Number of energy and force evaluations required for convergence of CI-NEB calculations in the H_2/Cu(110) example as a function of the number of degrees of freedom, increased by allowing a larger number of Cu-atoms to move. The performance of the all-images-evaluated (AIE) algorithm is presented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

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_{CI} = 0.01 eV/Å, T_{MEP} = 0.3 eV/Å, $T_{\mathrm{CIon}}^{\mathrm{GP}}$ = 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.

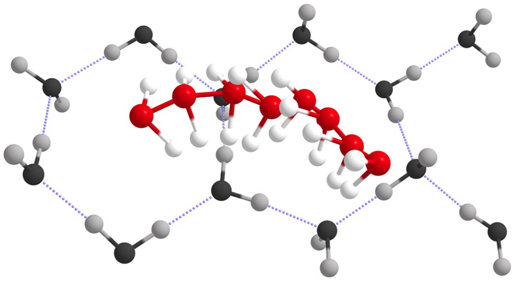


Figure 6. Minimum energy path for a diffusion hop of an H_2O admolecule on proton-disordered ice Ih(0001) surface calculated using the improved GP-NEB method. The O atom of the diffusing molecule is shown in red, and the molecules in the surface bilayer in grayscale. Lower bilayer molecules are not presented. 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 13 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]

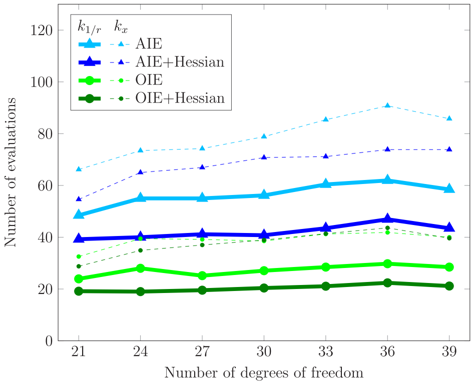


Figure 7. Number of energy and force evaluations required for convergence of CI-NEB calculations in the heptamer island benchmark with variants of the GP-NEB method. The average over the 13 different transitions is presented as a function of the number of degrees of freedom, increased by allowing a larger 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 is shown with green dots. The use of Hessian data at the initial and final state minima is indicated by darker color.

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 ($N_{\text{im}} = 7$) 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 ($T_{\text{CI}} = 0.01$ eV/Å, $T_{\text{MEP}} = 0.3$ eV/Å, $T_{\text{CIon}}^{\text{GP}} = 1$ 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).

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.

DOI: 10.1021/acs.jctc.9b00692
J. Chem. Theory Comput. 2019, 15, 6738−6751

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 $3N_{\rm m}$ input coordinates, this approach would mean quadratic scaling with respect to both the number of data points, N, and the number of moving atoms, $N_{\rm m}$. The construction of the matrix requires evaluations of $(N(1 + 3N_{\rm m}))^2$ covariances and the prediction of the whole gradient vector $N_{\rm m}$ derivatives of $N(1 + 3N_{\rm m})$ 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 $\mathcal{O}(N^2N_{\rm m}^2)$ time, and similarly, the prediction of the whole gradient vector can be done in $\mathcal{O}(NN_{\rm m}^2)$ 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.

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 dth coordinate of moving atom i in $\mathbf{x}$,

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

and with respect to both the d_1th coordinate of moving atom i_1 in $\mathbf{x}$ and d_2th coordinate of moving atom i_2 in $\mathbf{x}'$

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

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

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

and

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

where

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

and

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

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 $\sigma_{\rm m}$ is trivial, since $\sigma_{\rm m}^2$ 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:

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

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

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

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

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

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

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

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 dth coordinate of moving atom i in $\mathbf{x}$ is given by

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

and with respect to both the d_1th coordinate of moving atom i_1 in $\mathbf{x}$ and the d_2th coordinate of moving atom i_2 in $\mathbf{x}'$ by

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

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

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

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

and

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

■ ASSOCIATED CONTENT

Ⓢ Supporting Information

The Supporting Information is available free of charge on the ACS Publications website at DOI: 10.1021/acs.jctc.9b00692.

> 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)

■ AUTHOR INFORMATION

Corresponding Author

*E-mail: hj@hi.is.

ORCID

Olli-Pekka Koistinen: 0000-0002-0810-7369

Hannes Jónsson: 0000-0001-8285-5421

Funding

This work was financially supported by the Academy of Finland (Grant 278260; Flagship program: 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.

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

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 multi-component systems: applications to zinc oxide. *Phys. Rev. B: Condens. Matter Mater. Phys.* **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 is 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) Bartok, A. P.; Payne, M. C.; Kondor, R.; Csanyi, 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. *Modell. 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: Condens. Matter Mater. Phys.* **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. *arXiv.org* **2019**, 1903.08114.

Nudged Elastic Band Calculations Accelerated with Gaussian Process Regression Based on Inverse Interatomic Distances: Supporting Information

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

Matérn Covariance Functions

Modeling of systems with strong and quickly changing repulsive forces may be difficult with a stationary covariance function, where the characteristic length scale and magnitude stay the same throughout the coordinate space. As mentioned in the main article, one way to make a stationary covariance function more tolerant toward nonstationary effects is to loosen the assumptions of the smoothness of the modeled function conveyed through the smoothness properties of the covariance function. 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 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.

The Matérn covariance functions with smoothness parameter values $\nu = \frac{3}{2}$ and $\nu = \frac{5}{2}$, leading to once and twice differentiable sample functions, respectively, are given by

$$k_x^{\mathrm{M}-3/2}(\mathbf{x}, \mathbf{x}') = \sigma_{\mathrm{c}}^2 + \sigma_{\mathrm{m}}^2 \Big(\sqrt{3}\mathcal{D}_x(\mathbf{x}, \mathbf{x}') + 1\Big) \exp\Big(-\sqrt{3}\mathcal{D}_x(\mathbf{x}, \mathbf{x}')\Big) \tag{S1}$$

and

$$k_x^{\mathrm{M}-5/2}(\mathbf{x}, \mathbf{x}') = \sigma_{\mathrm{c}}^2 + \sigma_{\mathrm{m}}^2 \left(\frac{5}{3}\mathcal{D}_x^2(\mathbf{x}, \mathbf{x}') + \sqrt{5}\mathcal{D}_x(\mathbf{x}, \mathbf{x}') + 1\right) \exp\Big(-\sqrt{5}\mathcal{D}_x(\mathbf{x}, \mathbf{x}')\Big) \tag{S2}$$

The first, second, and third derivatives of Matérn-3/2 covariance function with respect to the square of difference measure $\mathcal{D}_x(\mathbf{x},\mathbf{x}')$ are given by

$$\frac{\partial k_x^{\mathrm{M-3/2}}(\mathbf{x},\mathbf{x}')}{\partial \mathcal{D}_x^2(\mathbf{x},\mathbf{x}')} = -\frac{3}{2}\sigma_{\mathrm{m}}^2 \exp\Big(-\sqrt{3}\mathcal{D}_x(\mathbf{x},\mathbf{x}')\Big) \tag{S3}$$

$$\frac{\partial^2 k_x^{\mathrm{M-3/2}}(\mathbf{x},\mathbf{x}')}{\partial (\mathcal{D}_x^2(\mathbf{x},\mathbf{x}'))^2} = \frac{3\sqrt{3}\sigma_{\mathrm{m}}^2}{4\mathcal{D}_x(\mathbf{x},\mathbf{x}')} \exp\Big(-\sqrt{3}\mathcal{D}_x(\mathbf{x},\mathbf{x}')\Big), \text{ when } \mathcal{D}_x(\mathbf{x},\mathbf{x}') > 0 \tag{S4}$$

and

$$\frac{\partial^3 k_x^{\mathrm{M-3/2}}(\mathbf{x},\mathbf{x}')}{\partial (\mathcal{D}_x^2(\mathbf{x},\mathbf{x}'))^3} = \frac{-3\sqrt{3}\sigma_{\mathrm{m}}^2}{8\mathcal{D}_x^3(\mathbf{x},\mathbf{x}')} \Big(\sqrt{3}\mathcal{D}_x(\mathbf{x},\mathbf{x}')+1\Big) \exp\Big(-\sqrt{3}\mathcal{D}_x(\mathbf{x},\mathbf{x}')\Big), \text{ when } \mathcal{D}_x(\mathbf{x},\mathbf{x}') > 0 \tag{S5}$$

and the corresponding derivatives of Matérn-5/2 by

$$\frac{\partial k_x^{\mathrm{M-5/2}}(\mathbf{x},\mathbf{x}')}{\partial \mathcal{D}_x^2(\mathbf{x},\mathbf{x}')} = -\frac{5}{6}\sigma_{\mathrm{m}}^2 \Big(\sqrt{5}\mathcal{D}_x(\mathbf{x},\mathbf{x}')+1\Big) \exp\Big(-\sqrt{5}\mathcal{D}_x(\mathbf{x},\mathbf{x}')\Big) \tag{S6}$$

$$\frac{\partial^2 k_x^{\mathrm{M-5/2}}(\mathbf{x},\mathbf{x}')}{\partial (\mathcal{D}_x^2(\mathbf{x},\mathbf{x}'))^2} = \frac{25}{12}\sigma_{\mathrm{m}}^2 \exp\Big(-\sqrt{5}\mathcal{D}_x(\mathbf{x},\mathbf{x}')\Big) \tag{S7}$$

and

$$\frac{\partial^3 k_x^{\mathrm{M-5/2}}(\mathbf{x},\mathbf{x}')}{\partial (\mathcal{D}_x^2(\mathbf{x},\mathbf{x}'))^3} = \frac{-25\sqrt{5}\sigma_{\mathrm{m}}^2}{24\mathcal{D}_x(\mathbf{x},\mathbf{x}')} \exp\Big(-\sqrt{5}\mathcal{D}_x(\mathbf{x},\mathbf{x}')\Big), \text{ when } \mathcal{D}_x(\mathbf{x},\mathbf{x}') > 0 \tag{S8}$$

In cases where these expressions are defined, the partial derivatives with respect to atom coordinates and length scale l are obtained by replacing derivatives of k_x with the above expressions in eqs 17–18 and 24–26. When $\mathcal{D}_x(\mathbf{x},\mathbf{x}') = 0$,

$$\frac{\partial^2 k_x^{\mathrm{M-3/2}}(\mathbf{x},\mathbf{x}')}{\partial x_{i_1,d_1}\partial x'_{i_2,d_2}} = -\frac{3}{2}\sigma_{\mathrm{m}}^2 \cdot \frac{\partial^2 \mathcal{D}_x^2(\mathbf{x},\mathbf{x}')}{\partial x_{i_1,d_1}\partial x'_{i_2,d_2}} \tag{S9}$$

$$\frac{\partial^2 k_x^{\mathrm{M-3/2}}(\mathbf{x},\mathbf{x}')}{\partial x_{i,d}\partial l} = 0 \tag{S10}$$

$$\frac{\partial^3 k_x^{\mathrm{M-3/2}}(\mathbf{x},\mathbf{x}')}{\partial x_{i_1,d_1}\partial x_{i_2,d_2}\partial l} = -\frac{3}{2}\sigma_{\mathrm{m}}^2 \cdot \frac{\partial^2 \mathcal{D}_x^2(\mathbf{x},\mathbf{x}')}{\partial x_{i_1,d_1}\partial x_{i_2,d_2}\partial l} \tag{S11}$$

and

$$\frac{\partial^3 k_x^{\mathrm{M-5/2}}(\mathbf{x},\mathbf{x}')}{\partial x_{i_1,d_1}\partial x_{i_2,d_2}\partial l} = -\frac{5}{6}\sigma_{\mathrm{m}}^2 \cdot \frac{\partial^2 \mathcal{D}_x^2(\mathbf{x},\mathbf{x}')}{\partial x_{i_1,d_1}\partial x_{i_2,d_2}\partial l} \tag{S12}$$

Figure S1 extends Figure 1 showing an energy curve for a pair of hydrogen atoms. In addition to GP approximations obtained with the squared exponential covariance function k_x, corresponding GP approximations are presented for Matérn covariance functions $k_x^{\mathrm{M-5/2}}$ and $k_x^{\mathrm{M-3/2}}$. As shown on the left, both Matérn-3/2 and Matérn-5/2 tolerate the long length scale better than the infinitely differentiable squared exponential covariance function. Even though the fit on the left looks good, stationarity causes problems also for $k_x^{\mathrm{M-5/2}}$ and $k_x^{\mathrm{M-3/2}}$, and optimization of the length scale leads to oscillations, although weaker than with k_x.

Figure S2 extends similarly Figure 2 showing a two-dimensional illustration where the vertical axis represents the distance of a pair of hydrogen atoms from a Cu(110) surface. In spite of quite a dense grid of observations, the squared exponential covariance function cannot recover from the oscillations caused by the high-gradient observations on the left. For Matérn-3/2 and

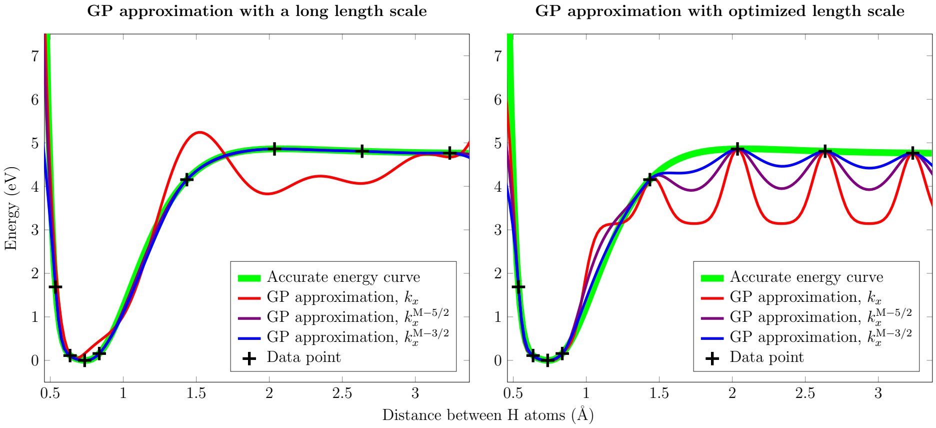


Figure S1. The thick green curve shows "true" energy as a function of distance between two hydrogen atoms. Training data for the GP models, marked with + signs, include accurate values for both energy and its first derivative with respect to the coordinate of the moving hydrogen atom. GP approximations obtained using stationary covariance functions with different smoothness properties are shown with red for the squared exponential covariance function k_x, violet for Matérn-5/2 covariance function $k_x^{\mathrm{M}-5/2}$, and blue for Matérn-3/2 covariance function $k_x^{\mathrm{M}-3/2}$. Left: GP approximations obtained with a long length scale (fixed hyperparameters: $\sigma_\mathrm{m} = 1.6$ eV, $l = 1$ Å). Right: GP approximations obtained with optimized hyperparameters (k_x: $\sigma_\mathrm{m} \approx 1.9$ eV, $l \approx 0.084$ Å; $k_x^{\mathrm{M}-5/2}$: $\sigma_\mathrm{m} \approx 1.9$ eV, $l \approx 0.17$ Å; $k_x^{\mathrm{M}-3/2}$: $\sigma_\mathrm{m} \approx 2.0$ eV, $l \approx 0.26$ Å).

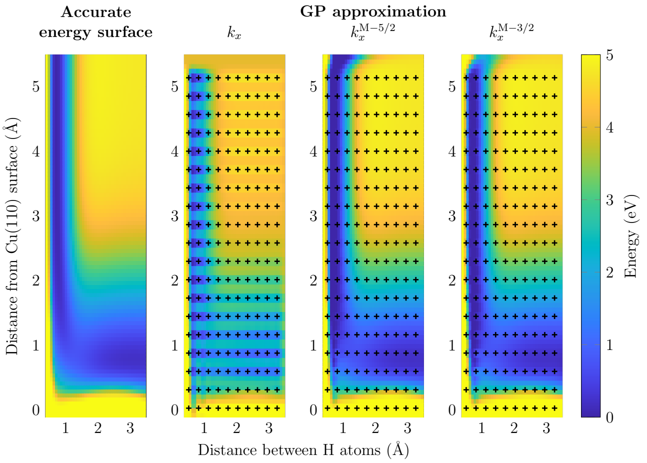


Figure S2. A two-dimensional cut through the potential energy surface for a pair of hydrogen atoms near a Cu(110) surface. The H–H molecular axis is parallel to the surface and perpendicular to the atom rows on the Cu(110) surface. The horizontal axis represents the distance between the two H atoms, and the vertical axis represents the distance between the H atoms and the Cu(110) surface. The leftmost graph presents "true" energy, and the three other graphs present GP approximations based on the grid of energy and atomic force evaluations shown with + signs when using optimized hyperparameters for stationary covariance functions with different smoothness properties. The squared exponential covariance function is notated as k_x and Matérn-5/2 and Matérn-3/2 covariance functions as $k_x^{\mathrm{M}-5/2}$ and $k_x^{\mathrm{M}-3/2}$, respectively.

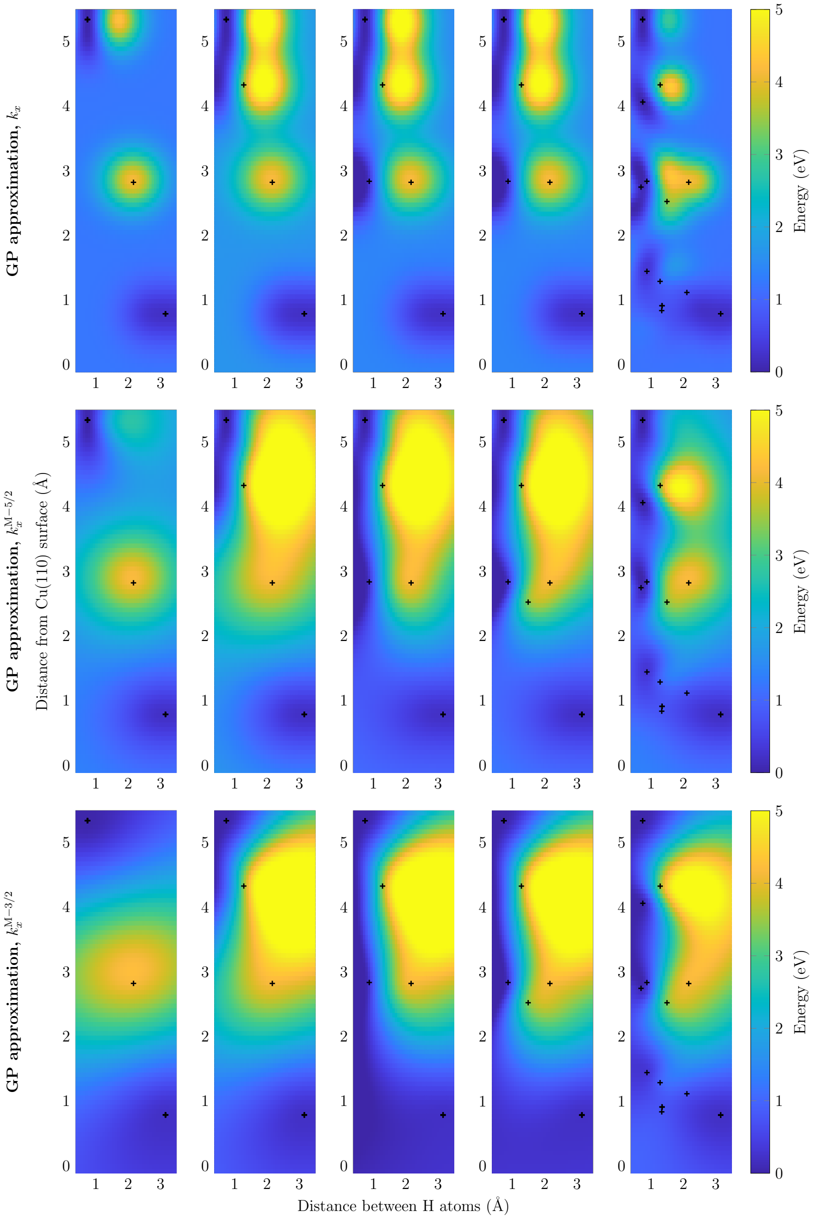


Figure S3. A two-dimensional cut through the potential energy surface for an H_2 molecule dissociating on a Cu(110) surface. The H–H molecular axis is parallel to the surface and perpendicular to the atom rows on the Cu(110) surface. The horizontal axis represents the distance between the two H atoms, and the vertical axis represents the distance between the H atoms and the Cu(110) surface. The three panels present GP approximations obtained using optimized hyperparameters for stationary covariance functions with different smoothness properties. The five training data sets, marked by + signs, include the same energy and force evaluations that are used in the first, second, third, fourth, and thirteenth GPR iteration of the improved GP-NEB algorithm using covariance function $k_{1/r}$; see Figure 3 in the main article.

Matérn-5/2, the numerous observations from the flat regions lengthen the optimized length scale, which allows smooth interpolation of those locations, but especially the lower left corner, where the H atoms are close to both the Cu(110) surface and each other, appears to be difficult to model correctly.

Figure S3 is an extension of the lower panel of Figure 3, where the training data sets do not include high-gradient observations from the repulsive regions. Again, Matérn-3/2 and Matérn-5/2 produce smoother interpolations than the squared exponential covariance function. Since Matérn-3/2 does not assume continuity of the second derivatives of the energy surface, it ignores the second derivative information included in the training data at the upmost and rightmost data points. For the same reason, the attractive forces acting on the H atoms are extrapolated farther to the repulsive region than when using $k_x^{\mathrm{M}-5/2}$ or k_x. As shown in Figure S4, an additional data point from the repulsive region would make interpolation of the training data set more difficult and lead to shorter length scale, although the effect is not as dramatic for Matérn-5/2 or Matérn-3/2.

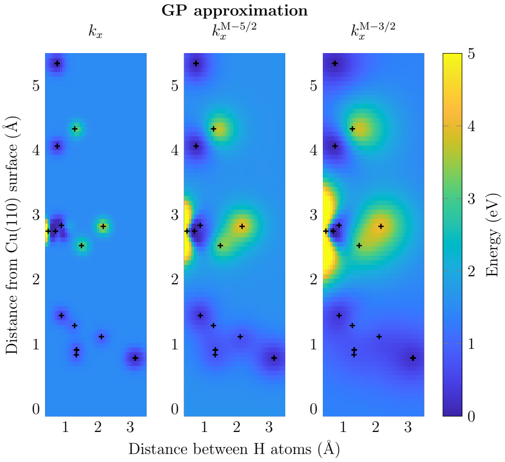


Figure S4. Illustrations of GP approximations based on stationary covariance functions k_x (left), $k_x^{\mathrm{M}-5/2}$ (middle), and $k_x^{\mathrm{M}-3/2}$ (right) corresponding to the rightmost graphs in Figure S3 after adding one high-gradient training data point near the left border of the graph and reoptimizing the hyperparameters.

Matérn Covariance Functions with Difference Measure $\mathcal{D}_{1/r}$

Similarly to the stationary squared exponential covariance function k_x, also covariance function $k_{1/r}$ based on the inverse-distance difference measure $\mathcal{D}_{1/r}$ can be made more flexible by feeding the difference measure to Matérn-3/2 or Matérn-5/2 covariance function:

$$k_{1/r}^{\mathrm{M}-3/2}(\mathbf{x},\mathbf{x}') = \sigma_{\mathrm{c}}^2 + \sigma_{\mathrm{m}}^2\Big(\sqrt{3}\mathcal{D}_{1/r}(\mathbf{x},\mathbf{x}') + 1\Big)\exp\Big(-\sqrt{3}\mathcal{D}_{1/r}(\mathbf{x},\mathbf{x}')\Big) \tag{S13}$$

$$k_{1/r}^{\mathrm{M}-5/2}(\mathbf{x},\mathbf{x}') = \sigma_{\mathrm{c}}^2 + \sigma_{\mathrm{m}}^2\left(\frac{5}{3}\mathcal{D}_{1/r}^2(\mathbf{x},\mathbf{x}') + \sqrt{5}\mathcal{D}_{1/r}(\mathbf{x},\mathbf{x}') + 1\right)\exp\Big(-\sqrt{5}\mathcal{D}_{1/r}(\mathbf{x},\mathbf{x}')\Big) \tag{S14}$$

The partial derivatives of covariance functions $k_{1/r}^{\mathrm{M}-3/2}$ and $k_{1/r}^{\mathrm{M}-5/2}$ can be presented with similar expressions as for $k_x^{\mathrm{M}-3/2}$ and $k_x^{\mathrm{M}-5/2}$, keeping in mind that the derivatives of $k_{1/r}^{\mathrm{M}-3/2}$ and $k_{1/r}^{\mathrm{M}-5/2}$

with respect to $\mathcal{D}_{1/r}^2$ are the same as the derivatives of $k_x^{\mathrm{M}-3/2}$ and $k_x^{\mathrm{M}-5/2}$ with respect to $\mathcal{D}_x^2$. The partial derivatives of the square of difference measure $\mathcal{D}_{1/r}$, required for these expressions, are presented in eqs 28–32.

Figure S5 repeats the results presented in Figure 7 for the improved GP-NEB method in the $\mathrm{H_2}$/Cu(110) example and shows also the corresponding results when using $k_{1/r}^{\mathrm{M}-5/2}$ or $k_{1/r}^{\mathrm{M}-3/2}$ instead of $k_{1/r}$ as the covariance function of the GP model. Except for the OIE results for $k_{1/r}^{\mathrm{M}-5/2}$, the Matérn variants increase the number of energy and force evaluations required for convergence. Matérn-3/2 variants using Hessian data are omitted, since $k_{1/r}^{\mathrm{M}-3/2}$ ignores second derivative information.

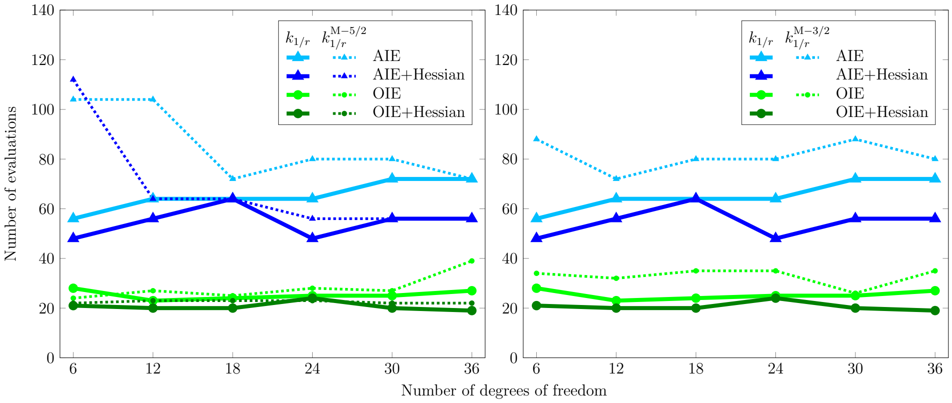


Figure S5. Number of energy and force evaluations required for convergence of CI-NEB calculations in the $\mathrm{H_2}$/Cu(110) example as a function of the number of degrees of freedom, increased by allowing a larger number of Cu atoms to move. The solid lines present results for the improved GP-NEB method when using covariance function $k_{1/r}$ obtained by feeding the inverse-distance difference measure $\mathcal{D}_{1/r}$ to the squared exponential covariance function. The dotted lines show the corresponding results when using covariance functions $k_{1/r}^{\mathrm{M}-5/2}$ (left) or $k_{1/r}^{\mathrm{M}-3/2}$ (right) obtained when the difference measure $\mathcal{D}_{1/r}$ is instead fed to Matérn-5/2 or Matérn-3/2 covariance function, respectively. 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.

Publication IV

Olli-Pekka Koistinen, Vilhjálmur Ásgeirsson, Aki Vehtari, and Hannes Jónsson. Minimum mode saddle point searches using Gaussian process regression with inverse-distance covariance function. Accepted for publication in *Journal of Chemical Theory and Computation*, 20 pages, December 2019.

Minimum mode saddle point searches using Gaussian process regression with inverse-distance covariance function

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

The minimum mode following method can be used to find saddle points on an energy surface by following a direction guided by the lowest curvature mode. Such calculations are often started close to a minimum on the energy surface to find out which transitions can occur from an initial state of the system, but it is also common to start from the vicinity of a first-order saddle point making use of an initial guess based on intuition or more approximate calculations. In systems where accurate evaluations of the energy and its gradient are computationally intensive, it is important to exploit the information of the previous evaluations to enhance the performance. Here, we show that the number of evaluations required for convergence to the saddle point can be significantly reduced by making use of an approximate energy surface obtained by a Gaussian process model based on inverse interatomic distances, evaluating accurate energy and gradient at the saddle point of the approximate surface and then correcting the model based on the new information. The performance of the method is tested with start points chosen randomly in the vicinity of saddle points for dissociative adsorption of an H_2 molecule on the Cu(110) surface and three gas phase chemical reactions.

1 Introduction

In systems characterized by a smooth potential energy surface, the transition state between the initial and final states of an atomic rearrangement event is, within the harmonic approximation to transition state theory, placed using information about a first-order saddle point on the energy surface, i.e., a location with zero gradient and exactly one negative eigenvalue of the Hessian matrix. Finding first-order saddle points is then the essential task when identifying the mechanisms and estimating the rates of transitions. The transition state is taken to be a hyperplane going through the saddle point with normal parallel to the eigenvector corresponding to the negative eigenvalue.

In chain-of-states methods, such as the nudged elastic band method,[1,2] saddle points are found by calculating a minimum energy path between the initial and final states and identifying the energy maximum along the path. There, both the initial and final states of the transition are specified. In another type of algorithms, only the initial state is specified and the saddle point found by climbing up the energy surface without specifying the final state of the transition. Such calculations are often started from the vicinity of the initial state minimum to find out which transitions can occur, but it is also common to start from somewhere close to the saddle point with an initial guess obtained from intuition or from approximate minimum energy path calculations.[3,4] Early algorithms of this sort required the evaluation of the full Hessian, the matrix of the second derivatives of the energy with respect to the coordinates, and calculation of all the eigenvalues and eigenvectors (see ref 5 for a review). In a more efficient formulation, the minimum mode following method, only the eigenvector corresponding to the lowest eigenvalue is found and used to guide the search for the saddle point(s) without a need to evaluate the Hessian matrix.[6–8]

In this article, we choose to find the minimum mode using the dimer method.[6,9–11] A dimer is here a pair of points in a configuration space, separated by a small fixed distance. The dimer is first rotated around its midpoint to find the orientation that gives the lowest total energy of the two configurations. This gives the direction of the lowest curvature mode of the Hessian, the minimum mode.[12] The dimer is then translated toward the saddle point by reversing the force (negative energy gradient) component in this direction. The movements are based only on the energy and the gradient of the energy and thus do not require calculation of the Hessian matrix.

In systems where accurate evaluations of energy and its gradient are computationally expensive, it is important to exploit the information in previous evaluations to enhance the performance. Here, we show that Gaussian process (GP) regression[13–16] can be used to significantly reduce the number of evaluations required for convergence to saddle points. The basic scheme is similar to the one used for nudged elastic band calculations in the GP-NEB method:[17–19] a regular minimum mode following calculation is performed to find a saddle point on an approximate surface obtained by a Gaussian process model, accurate energy and gradient are then evaluated at that point, and the model is subsequently refined based on the new evaluations. If no information about the energy surface is available in the beginning, it is useful to perform initial rotations with accurate evaluations before starting to translate the dimer. We show that GP regression can be used also to reduce the number of evaluations required for finding the lowest curvature mode in this initial rotation phase.

A similar general scheme for saddle point searches starting from a configuration close to a saddle point has been presented by Denzel and Kästner, who use a stationary Matérn covariance function to build the GP model.[20] Here, we use a more expressive covariance function based on inverted interatomic distances, coupled with a robust stopping criterion, as suggested in ref 19 and compare the performance to stationary covariance functions. The inverse-distance covariance function makes the method more robust especially when the calculation is started far from the saddle point where the atomic forces are large.

The performance of the GP-dimer method is tested with start points up to 3 Å from saddle points for dissociative adsorption of an H_2 molecule on the Cu(110) surface and three gas phase chemical reactions. With the largest start distances, the number of energy and gradient evaluations is found to be reduced by an order of magnitude compared to the regular dimer method.

2 Dimer method

In this section, we review the principles of the dimer method for finding the minimum mode[6,9–11] and present details for two variants of the algorithm, here referred to as CG-dimer[10] and LBFGS-dimer.[11] They are used as references for comparing the performance with our GP-dimer method. The LBFGS-dimer algorithm is used also as a part of the GP-dimer method as described in the following section.

A dimer is defined as a pair of points in a configuration space, referred to as image 1, $\mathbf{R}_1$, and image 2, $\mathbf{R}_2$. The small distance between $\mathbf{R}_1$ and $\mathbf{R}_2$ is kept constant, and half of this distance is referred to as the dimer separation, $\Delta_\mathbf{R}$ (here 10^{-2} Å as recommended in ref 9). The middle point of the dimer is denoted by $\mathbf{R}_0$, and the orientation vector $\hat{\mathbf{N}}$ is a unit vector that points from $\mathbf{R}_0$ toward $\mathbf{R}_1$. The dimer energy is defined as the sum $E_1 + E_2$, where E_1 and E_2 denote the energy of the system at $\mathbf{R}_1$ and $\mathbf{R}_2$, respectively. The direction of lowest curvature of energy at $\mathbf{R}_0$ corresponds to the orientation of minimum dimer energy, which is obtained by rotating the dimer around $\mathbf{R}_0$ so that the rotational force is zeroed. Denoting the force (negative energy gradient) acting on $\mathbf{R}_i$ by $\mathbf{F}_i$ and the component of $\mathbf{F}_i$ perpendicular to the dimer by $\mathbf{F}_i^\perp = \mathbf{F}_i - (\mathbf{F}_i \cdot \hat{\mathbf{N}})\hat{\mathbf{N}}$, the scaled rotational force acting on $\mathbf{R}_1$ is defined by

$$\mathbf{F}_\mathrm{rot} = (\mathbf{F}_1^\perp - \mathbf{F}_2^\perp)/\Delta_\mathbf{R}. \tag{1}$$

As suggested by Olsen et al.,[9] it is more efficient to evaluate the force at the middle point $\mathbf{R}_0$ instead of $\mathbf{R}_2$ and extrapolate the force at $\mathbf{R}_2$ as $\mathbf{F}_2 = 2\mathbf{F}_0 - \mathbf{F}_1$.

Each rotation iteration is performed within a plane spanned by unit vectors $\hat{\mathbf{N}}$ and $\hat{\mathbf{\Omega}}$, where the steepest descent direction of rotation for image 1 is $\hat{\mathbf{\Omega}} = \mathbf{F}_\mathrm{rot}/||\mathbf{F}_\mathrm{rot}||$. In CG-dimer and LBFGS-dimer, $\hat{\mathbf{\Omega}}$ is modified based on previous rotation iterations according to nonlinear conjugate gradient[21,22] or limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS)[23,24] algorithms, respectively. According to the approach of Heyden et al.,[10] a rough estimate for the optimal rotational angle is first calculated based on $\mathbf{F}_0$ and $\mathbf{F}_1$ as

$$\omega^* = \frac{1}{2}\arctan\frac{(\mathbf{F}_1 - \mathbf{F}_0)\cdot\hat{\mathbf{\Omega}}}{\Delta_\mathbf{R}\,|C|}, \tag{2}$$

where $C = (\mathbf{F}_0 - \mathbf{F}_1)\cdot\hat{\mathbf{N}}/\Delta_\mathbf{R}$ is the curvature of the energy along the dimer. After the preliminary rotation of ω^*, the orientation vector of the dimer is given by

$$\hat{\mathbf{N}}^* = \hat{\mathbf{N}}\cos\omega^* + \hat{\mathbf{\Omega}}\sin\omega^* \tag{3}$$

and the rotation direction by

$$\hat{\mathbf{\Omega}}^* = -\hat{\mathbf{N}}\sin\omega^* + \hat{\mathbf{\Omega}}\cos\omega^*. \tag{4}$$

After evaluating the force $\mathbf{F}_1^*$ at $\mathbf{R}_1^* = \mathbf{R}_0 + \Delta_\mathbf{R}\hat{\mathbf{N}}^*$, the optimal rotational angle based on a local quadratic approximation to the energy surface is given by

$$\omega = \begin{cases} \frac{1}{2}\arctan\frac{b_1}{a_1}, & \text{if } \frac{b_1}{a_1} \geq 0 \\ \frac{1}{2}\arctan\frac{b_1}{a_1} + \frac{\pi}{2}, & \text{if } \frac{b_1}{a_1} < 0, \end{cases} \tag{5}$$

where

$$b_1 = (\mathbf{F}_0 - \mathbf{F}_1)\cdot\hat{\mathbf{\Omega}}/\Delta_\mathbf{R} \tag{6}$$

and

$$a_1 = \frac{b_1\cos(2\omega^*) - (\mathbf{F}_0 - \mathbf{F}_1^*)\cdot\hat{\mathbf{\Omega}}^*/\Delta_\mathbf{R}}{\sin(2\omega^*)}. \tag{7}$$

The orientation vector of the dimer after the rotation is then given by

$$\hat{\mathbf{N}}^{\text{new}} = \hat{\mathbf{N}} \cos\omega + \hat{\mathbf{\Omega}} \sin\omega \tag{8}$$

and the new location of image 1 by

$$\mathbf{R}_1^{\text{new}} = \mathbf{R}_0 + \Delta_{\mathbf{R}} \hat{\mathbf{N}}^{\text{new}}. \tag{9}$$

The rotation direction in the end of the rotation, needed for the following rotation iteration in CG-dimer, is given by

$$\hat{\mathbf{\Omega}}_{\text{end}} = -\hat{\mathbf{N}} \sin\omega + \hat{\mathbf{\Omega}} \cos\omega. \tag{10}$$

Here, the rotation iterations are stopped if the preliminary rotational angle ω^* is estimated to be below 5°,[11] if the actual rotational angle ω is below this threshold, or if a prescribed maximum number of consecutive rotation iterations is reached. In the two latter cases, the curvature of energy along the new orientation vector $\hat{\mathbf{N}}^{\text{new}}$ can be estimated as

$$C^{\text{new}} \approx C + a_1(\cos(2\omega) - 1) + b_1 \sin(2\omega). \tag{11}$$

After the rotation phase, the middle point of the dimer is translated in order to advance toward the saddle point. The translational force is obtained by inverting the component of $\mathbf{F}_0$ parallel to the dimer:

$$\mathbf{F}_{\text{trans}} = \mathbf{F}_0 - 2\mathbf{F}_0^{||}, \tag{12}$$

where $\mathbf{F}_0^{||} = (\mathbf{F}_0 \cdot \hat{\mathbf{N}})\hat{\mathbf{N}}$. This allows the dimer to climb upward on the energy surface in the direction of the minimum mode while moving toward lower energy in directions perpendicular to the minimum mode. In CG-dimer and LBFGS-dimer, also the translations are modified according to conjugate gradient and L-BFGS algorithms, respectively. If the curvature along the dimer is positive, the dimer is assumed to be in a convex region where all eigenvalues of the Hessian matrix are positive. In this case, a step of a predefined length is taken in the opposite direction of $\mathbf{F}_0^{||}$ to make the dimer climb up from the energy basin as quickly as possible. Here, this step length is set to 0.1 Å, which is also the maximum step length for the translation iterations.[9] The calculation is considered to have converged when the maximum component of force $\mathbf{F}_0$ at the middle point of the dimer is below a threshold T_0, which is here set to $T_0 = 0.01$ eV/Å.

2.1 CG-dimer

The first of the two reference algorithms, referred to here as CG-dimer, follows mainly the details presented by Heyden et al.[10] In this algorithm, only one rotation iteration, if any, is performed between translations. Thus, each rotation phase includes two or three energy and force evaluations. Since the initial orientation of the dimer is chosen randomly in our test cases, a larger maximum number of rotations, equal to the number of degrees of freedom in the system, is used in the first rotation phase to stabilize the algorithm.

In CG-dimer, we apply separate nonlinear conjugate gradient algorithms[21,22] to choose the rotational plane and translational search direction, as suggested previously.[6] Given a rotation direction $\hat{\mathbf{\Omega}}$, the rotation proceeds as presented above. For translations, a preliminary step is first taken and the middle point of the dimer moved to

$$\mathbf{R}_0^* = \mathbf{R}_0 + \frac{\hat{\mathbf{\Gamma}} \cdot \mathbf{F}_{\text{trans}}}{2\,|C|} \hat{\mathbf{\Gamma}}, \tag{13}$$

where $\hat{\mathbf{\Gamma}}$ is a unit vector parallel to the search direction.[10] After evaluating the force $\mathbf{F}_0^*$ at $\mathbf{R}_0^*$, the middle point of the dimer is then moved to the estimated zero point of the translational force component parallel to the search direction:

$$\mathbf{R}_0^{\mathrm{new}} = \mathbf{R}_0 - \frac{\hat{\mathbf{\Gamma}} \cdot \mathbf{F}_{\mathrm{trans}}}{\hat{\mathbf{\Gamma}} \cdot (\mathbf{F}_{\mathrm{trans}}^* - \mathbf{F}_{\mathrm{trans}})}(\mathbf{R}_0^* - \mathbf{R}_0), \tag{14}$$

where $\mathbf{F}_{\mathrm{trans}}^* = \mathbf{F}_0^* - 2(\mathbf{F}_0^* \cdot \hat{\mathbf{N}})\hat{\mathbf{N}}$.

In the conjugate gradient algorithm for the translations, the search direction $\hat{\mathbf{\Gamma}}$ is parallel to a conjugated force vector $\mathbf{\Gamma}$, which is a linear combination of the current and previous translational force vectors. $\mathbf{\Gamma}$ can be expressed recursively as

$$\mathbf{\Gamma} = \mathbf{F}_{\mathrm{trans}} + \beta \mathbf{\Gamma}^{\mathrm{old}}, \tag{15}$$

where $\mathbf{\Gamma}^{\mathrm{old}}$ is the conjugated force vector in the previous translation iteration and the coefficient β_{trans} is here given by the Polak-Ribière formula:[22]

$$\beta_{\mathrm{trans}} = \max\left\{0, \frac{(\mathbf{F}_{\mathrm{trans}} - \mathbf{F}_{\mathrm{trans}}^{\mathrm{old}}) \cdot \mathbf{F}_{\mathrm{trans}}}{\mathbf{F}_{\mathrm{trans}}^{\mathrm{old}} \cdot \mathbf{F}_{\mathrm{trans}}^{\mathrm{old}}}\right\}, \tag{16}$$

where $\mathbf{F}_{\mathrm{trans}}^{\mathrm{old}}$ is the previous translational force vector. In the first iteration, $\mathbf{\Gamma}$ is set equal to $\mathbf{F}_{\mathrm{trans}}$. As noted previously,[6] increasing translational force in the search direction would lead to a step backward against the search direction, which indicates that the dimer may still be in a convex area in spite of negative estimated curvature C in the direction of the dimer. In this case, a step of a predefined length (here 0.1 Å) is taken in the search direction. Without further restrictions, however, a negative step against the search direction may occur also if the search direction itself is opposite to the current translational force vector. This would as well trigger the predefined step and might lead to a trap where the dimer bounces between two locations. To prevent this kind of situation, we set β_{trans} to zero when the correction vector $\beta_{\mathrm{trans}}\mathbf{\Gamma}^{\mathrm{old}}$ in eq 15 becomes longer than the current translational force vector $\mathbf{F}_{\mathrm{trans}}$. In addition, we reset the memory of conjugate directions when the number of conjugated iterations reaches the number of degrees of freedom in the system or if a predefined step length is used due to positive C, negative step against the search direction, or excessive step length.

Analogously to the conjugate gradient algorithm described above for the translations, the rotation direction $\hat{\mathbf{\Omega}}$ is parallel to a conjugated force vector $\mathbf{\Omega}$ defined recursively based on the current rotational force vector $\mathbf{F}_{\mathrm{rot}}$, the previous rotational force vector $\mathbf{F}_{\mathrm{rot}}^{\mathrm{old}}$, and the previous conjugated force vector $\mathbf{\Omega}^{\mathrm{old}}$. The only difference is that $\mathbf{\Omega}^{\mathrm{old}}$ needs to be rotated on the previous rotational plane to be aligned with $\hat{\mathbf{\Omega}}_{\mathrm{end}}^{\mathrm{old}}$, which is the rotation direction in the end of the previous iteration (eq 10).[6] Thus, the recursive expression for $\mathbf{\Omega}$ is given by

$$\mathbf{\Omega} = \mathbf{F}_{\mathrm{rot}} + \beta_{\mathrm{rot}} ||\mathbf{\Omega}^{\mathrm{old}}|| \hat{\mathbf{\Omega}}_{\mathrm{end}}^{\mathrm{old}}, \tag{17}$$

where

$$\beta_{\mathrm{rot}} = \max\left\{0, \frac{(\mathbf{F}_{\mathrm{rot}} - \mathbf{F}_{\mathrm{rot}}^{\mathrm{old}}) \cdot \mathbf{F}_{\mathrm{rot}}}{\mathbf{F}_{\mathrm{rot}}^{\mathrm{old}} \cdot \mathbf{F}_{\mathrm{rot}}^{\mathrm{old}}}\right\}. \tag{18}$$

The coefficient β_{rot} is set to zero when the correction vector $\beta_{\mathrm{rot}} ||\mathbf{\Omega}^{\mathrm{old}}|| \hat{\mathbf{\Omega}}_{\mathrm{end}}^{\mathrm{old}}$ in eq 17 becomes longer than the current rotational force vector $\mathbf{F}_{\mathrm{rot}}$, and the memory of rotational conjugate directions is reset when the number of conjugated rotation iterations reaches the number of degrees of freedom in the system or if rotational convergence is reached.

2.2 LBFGS-dimer

The second reference algorithm, referred to here as LBFGS-dimer, follows mainly the details presented by Kästner and Sherwood.[11] In this algorithm, the rotations are continued until convergence unless the maximum of 10 consecutive rotation iterations is reached. If the number of degrees of freedom is less than 10, we use this number as the maximum. To reduce the number of evaluations between the consecutive rotation iterations to one, the force $\mathbf{F}_1^{\text{new}}$ at the new location of image 1 is estimated as

$$\mathbf{F}_1^{\text{new}} \approx \frac{\sin(\omega^* - \omega)}{\sin\omega^*}\mathbf{F}_1 + \frac{\sin\omega}{\sin\omega^*}\mathbf{F}_1^* + \left(1 - \cos\omega - \sin\omega\tan\frac{\omega^*}{2}\right)\mathbf{F}_0. \tag{19}$$

In LBFGS-dimer, the rotational plane and translational search direction are chosen using separate limited-memory BFGS algorithms.[23,24] Given a rotation direction $\hat{\boldsymbol{\Omega}}$, the rotation proceeds similarly as in CG-dimer. For translations, the L-BFGS algorithm gives also a step length in addition to the search direction, and thus no preliminary step is needed.

The L-BFGS algorithm approximates an inverse Hessian matrix implicitly based on information stored from previous iterations. The memory of L-BFGS includes displacement vectors $\boldsymbol{\delta}_{\mathbf{x}}^i, i = 1, 2, \ldots, M$, between the locations and $\boldsymbol{\delta}_{\mathbf{F}}^i, i = 1, 2, \ldots, M$, between the effective forces in the i^{th} and $(i-1)^{\text{th}}$ last iteration counting backward from the current iteration. The size of the memory, M, is here limited to the number of degrees of freedom in the system, and the inverse Hessian is initialized to an identity matrix scaled by $\lambda = (\boldsymbol{\delta}_{\mathbf{F}}^1 \cdot \boldsymbol{\delta}_{\mathbf{x}}^1)/||\boldsymbol{\delta}_{\mathbf{F}}^1||^2$.[24] If the memory is empty, the scaling factor is set to $\lambda = 0.01$ Å^2/eV. The optimal displacement vector $\boldsymbol{\delta}_{\mathbf{x}}$ for the current iteration is obtained by the following recursive procedure,[23] where $\boldsymbol{\Psi}$ is initialized to the effective force vector:

```
For $i = 1, 2, \ldots, M$ :
    Set $\alpha^i \leftarrow \frac{\boldsymbol{\Psi} \cdot \boldsymbol{\delta}_{\mathbf{x}}^i}{\boldsymbol{\delta}_{\mathbf{F}}^i \cdot \boldsymbol{\delta}_{\mathbf{x}}^i}$.
    Set $\boldsymbol{\Psi} \leftarrow \boldsymbol{\Psi} - \alpha^i \boldsymbol{\delta}_{\mathbf{F}}^i$.
Set $\boldsymbol{\delta}_{\mathbf{x}} \leftarrow \lambda\boldsymbol{\Psi}$.
For $i = M, M-1, \ldots, 1$ :
    Set $\boldsymbol{\delta}_{\mathbf{x}} \leftarrow \boldsymbol{\delta}_{\mathbf{x}} + \left(\alpha^i - \frac{\boldsymbol{\delta}_{\mathbf{F}}^i \cdot \boldsymbol{\delta}_{\mathbf{x}}}{\boldsymbol{\delta}_{\mathbf{F}}^i \cdot \boldsymbol{\delta}_{\mathbf{x}}^i}\right)\boldsymbol{\delta}_{\mathbf{x}}^i$.
```

In the L-BFGS algorithm for the translations, $\boldsymbol{\delta}_{\mathbf{x}}^i$ are displacements of the middle point $\mathbf{R}_0$ and the effective force is the translational force $\mathbf{F}_{\text{trans}}$. The new location of $\mathbf{R}_0$ is simply given by $\mathbf{R}_0^{\text{new}} + \boldsymbol{\delta}_{\mathbf{x}}$. The memory of L-BFGS is reset, if a predefined step length is used due to positive C or excessive step length.

For the rotations, $\boldsymbol{\delta}_{\mathbf{x}}^i$ are given by changes of the orientation vector $\hat{\mathbf{N}}$ during previous rotation iterations and the effective force is the rotational force $\mathbf{F}_{\text{rot}}$. After estimating $\boldsymbol{\delta}_{\mathbf{x}}$ according to the recursive procedure described above, the rotation direction $\hat{\boldsymbol{\Omega}}$ is given by a unit vector parallel to

$$\boldsymbol{\delta}_{\mathbf{x}}^{\perp} = \boldsymbol{\delta}_{\mathbf{x}} - (\boldsymbol{\delta}_{\mathbf{x}} \cdot \hat{\mathbf{N}})\hat{\mathbf{N}}, \tag{20}$$

which is the component of $\boldsymbol{\delta}_{\mathbf{x}}$ perpendicular to the dimer. The memory of the L-BFGS algorithm for rotations is reset after each translation step.

3 GP-dimer method

In this section, the GP-dimer method is described. There, each iteration involves the energy surface being modeled using Gaussian process regression, dimer calculations performed on the approximate surface, and the GP model then refined after evaluating the accurate energy and force at the saddle point determined on the approximate surface. The method can be seen as a surface walking version of the GP-NEB method.[17–19] The dimer calculations on the approximated surface are performed using LBFGS-dimer[11] with some modifications.

3.1 Gaussian process regression

A Gaussian process[13–16] model defines 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 locations $\mathbf{X} = [\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(N)}]^\mathsf{T}$ as a multivariate Gaussian $p(\mathbf{f}) = \mathcal{N}(\mathbf{m}, K(\mathbf{X}, \mathbf{X}))$, where $\mathbf{m} = [m(\mathbf{x}^{(1)}), m(\mathbf{x}^{(2)}), \ldots, m(\mathbf{x}^{(N)})]^\mathsf{T}$ is defined by mean function $m(\mathbf{x})$ and the notation $K(\mathbf{X}, \mathbf{X}')$ stands for a covariance matrix with elements $K_{ij} = k(\mathbf{x}^{(i)}, \mathbf{x}'^{(j)})$ defined by covariance function $k(\mathbf{x}, \mathbf{x}')$. In the applications of the GP-dimer method presented here, the energy surface is modeled as a function of a $3N_\mathrm{m}$-dimensional coordinate vector

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

including the coordinates for moving atoms $1, 2, \ldots, N_\mathrm{m} \in A_\mathrm{m}$. The system may also involve a set of atoms with fixed coordinates, denoted by A_f, but here those atoms are taken into account in the GP model only if some of the moving atoms have been within the radius of 5 Å from the frozen atom during the GP-dimer algorithm. As suggested in ref 19, the prior probability model of the energy surface is defined here as a GP with mean function $m(\mathbf{x}) = 0$ and covariance function

$$k_{1/r}(\mathbf{x}, \mathbf{x}') = \sigma_\mathrm{c}^2 + \sigma_\mathrm{m}^2 \exp\left(-\frac{1}{2} \sum_{i \in A_\mathrm{m}} \sum_{\substack{j \in A_\mathrm{m}, j > i \\ \vee \\ j \in A_\mathrm{f}}} \frac{\left(\frac{1}{r_{i,j}(\mathbf{x})} - \frac{1}{r_{i,j}(\mathbf{x}')} \right)^2}{l_{\phi(i,j)}^2} \right), \tag{21}$$

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. With the inverse-distance formulation, a displacement of an atom toward or away from another atom results in a larger drop when the two atoms are closer to each other, which makes it easier to model large repulsive forces.

Weakly informative prior distributions $p(\sigma_\mathrm{m}) = \mathcal{N}(0, \max\{1 \text{ eV}^2, (\Delta_\mathbf{y}/3)^2\})$ and $p(l_\psi) = \mathcal{N}(0, \max\{1 \text{ Å}^{-2}, (\Delta_\mathbf{X}/3)^2\})$ are set for the magnitude σ_m and length scales $l_\psi, \psi = 1, 2, \ldots, N_\phi$, with $\Delta_\mathbf{y}$ representing the range of energy values in the training data set and $\Delta_\mathbf{X}$ representing the maximum difference between the data points based on difference measure

$$\mathcal{D}_{1/r}(\mathbf{x}, \mathbf{x}') = \sqrt{\sum_{i \in A_\mathrm{m}} \sum_{\substack{j \in A_\mathrm{m}, j > i \\ \vee \\ j \in A_\mathrm{f}}} \left(\frac{1}{r_{i,j}(\mathbf{x})} - \frac{1}{r_{i,j}(\mathbf{x}')} \right)^2}. \tag{22}$$

The constant term σ_c^2 corresponds to a prior variance for an unknown constant mean function and is set to the square of the mean of the observed energy values but no lower than 1 eV^2. The only differences from the model suggested for the GP-NEB method in ref 19 are the lower limits for the constant term and for the variances of the prior distributions of the hyperparameters. Since the initial training data set in the GP-dimer method is focused around one start point, small $\Delta_{\mathbf{X}}$ and $\Delta_{\mathbf{y}}$ would lead to unnecessarily restrictive priors for the magnitude σ_{m} and length scales l_ψ in the beginning if no lower limits for the variances were used.

For comparisons, alternative GP models with stationary covariance functions are also implemented for the GP-dimer method. Following the notation of ref 19, we define the squared exponential covariance function as

$$k_x(\mathbf{x},\mathbf{x}') = \sigma_{\mathrm{c}}^2 + \sigma_{\mathrm{m}}^2 \exp\left(-\frac{||\mathbf{x}-\mathbf{x}'||^2}{2l^2}\right) \tag{23}$$

and the Matérn-5/2 covariance function as

$$k_x^{\mathrm{M}-5/2}(\mathbf{x},\mathbf{x}') = \sigma_{\mathrm{c}}^2 + \sigma_{\mathrm{m}}^2 \left(1+\frac{\sqrt{5}||\mathbf{x}-\mathbf{x}'||}{l}+\frac{5||\mathbf{x}-\mathbf{x}'||^2}{3l^2}\right)\exp\left(-\frac{\sqrt{5}||\mathbf{x}-\mathbf{x}'||}{l}\right). \tag{24}$$

Since the dimer method relies on the curvature properties of the energy surface, Matérn covariance functions with a lower smoothness parameter ($\nu < 2$) are not good choices for this application. The priors of the hyperparameters are defined similarly as for $k_{1/r}$, but the prior variance of the length scale is based on the regular distance in the $3N_{\mathrm{m}}$-dimensional coordinate space.

The evaluations of energy and force (negative energy gradient) are regarded as accurate up to floating point presentation accuracy, and thus Gaussian noise with a small variance is assumed to be included in both energy (noise variance $\sigma^2 = 10^{-8}$ eV^2) and force evaluations (noise variance $\sigma_{\mathrm{d}}^2 = 10^{-8}$ eV^2/Å^2) to avoid numerical problems. Given a training data set $\{\mathbf{X},\mathbf{y}\}$, where $\mathbf{y} = [y^{(1)}, y^{(2)}, \ldots, y^{(N)}]^{\mathsf{T}}$ includes evaluated energy values from N locations $\mathbf{X}$, and a noise covariance matrix $\mathbf{\Sigma} = \sigma^2\mathbf{I}_N$ with $\mathbf{I}_N$ denoting an identity matrix, the hyperparameters $\boldsymbol{\theta} = \{\sigma_{\mathrm{m}}, l_1, l_2, \ldots, l_{N_\phi}\}$ can be optimized by 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 $p(\boldsymbol{\theta}) = p(\sigma_{\mathrm{m}})\prod_{\psi=1}^{N_\phi} p(l_\psi)$ and

$$p(\mathbf{y}\,|\,\mathbf{X},\boldsymbol{\theta}) = \mathcal{N}(\mathbf{y}\,|\,\mathbf{0}, K(\mathbf{X},\mathbf{X}) + \mathbf{\Sigma}) \tag{25}$$

is the marginal likelihood of $\boldsymbol{\theta}$. The GP approximation for the energy $f(\mathbf{x}^*)$ at any location $\mathbf{x}^*$ is then obtained by GP regression as the mean of the posterior predictive distribution of $f(\mathbf{x}^*)$ conditional on the optimized hyperparameters $\boldsymbol{\theta}$,

$$\mathrm{E}[f(\mathbf{x}^*)\,|\,\mathbf{y},\mathbf{X},\boldsymbol{\theta}] = K(\mathbf{x}^*,\mathbf{X})(K(\mathbf{X},\mathbf{X})+\mathbf{\Sigma})^{-1}\mathbf{y}, \tag{26}$$

and the approximation for the partial derivative of the energy with respect to coordinate $x_{i,d}^*$ is given by

$$\mathrm{E}\left[\frac{\partial f(\mathbf{x}^*)}{\partial x_{i,d}^*}\,\middle|\,\mathbf{y},\mathbf{X},\boldsymbol{\theta}\right] = \frac{\partial K(\mathbf{x}^*,\mathbf{X})}{\partial x_{i,d}^*}(K(\mathbf{X},\mathbf{X})+\mathbf{\Sigma})^{-1}\mathbf{y}, \tag{27}$$

where the elements of $\partial K(\mathbf{x}^*,\mathbf{X})/\partial x_{i,d}^*$ are obtained by differentiating the covariance function.

The derivatives of the covariance function are needed also when including the force evaluations in the training data set.[25–28] When $\mathbf{y}$ is extended to include partial derivatives of f (components of negative force), the training covariance matrix $K(\mathbf{X},\mathbf{X})$ and covariance vector $K(\mathbf{x}^*,\mathbf{X})$ are extended correspondingly to include prior covariances between the energy and derivative values

$$\mathrm{Cov}\left[\frac{\partial f(\mathbf{x})}{\partial x_{i,d}}, f(\mathbf{x}')\right] = \frac{\partial}{\partial x_{i,d}}\mathrm{Cov}\big[f(\mathbf{x}), f(\mathbf{x}')\big] = \frac{\partial k(\mathbf{x},\mathbf{x}')}{\partial x_{i,d}} \tag{28}$$

and the covariances between the derivatives

$$\mathrm{Cov}\left[\frac{\partial f(\mathbf{x})}{\partial x_{i_1,d_1}}, \frac{\partial f(\mathbf{x}')}{\partial x'_{i_2,d_2}}\right] = \frac{\partial^2}{\partial x_{i_1,d_1}\partial x'_{i_2,d_2}}\mathrm{Cov}\big[f(\mathbf{x}), f(\mathbf{x}')\big] = \frac{\partial^2 k(\mathbf{x},\mathbf{x}')}{\partial x_{i_1,d_1}\partial x'_{i_2,d_2}}, \tag{29}$$

and the noise covariance matrix $\mathbf{\Sigma}$ is extended to include the noise variances for the force evaluations (σ_{d}^2) on the diagonal. Expressions for the derivatives of covariance functions $k_{1/r}$, k_x and $k_x^{\mathrm{M}-5/2}$ with respect to the atom coordinates $x_{i,d}$ and the hyperparameters $\boldsymbol{\theta}$ can be found in ref 19. The latter are useful in the hyperparameter optimization, which is here performed with the scaled conjugate gradient algorithm[29] implemented in the GPstuff toolbox.[30]

3.2 Algorithm description

If no information about the energy surface or the minimum energy orientation of the dimer is available in the beginning, it is useful to perform initial rotations with accurate evaluations to find the lowest curvature mode before starting to translate the dimer. These rotations can be done through a regular rotation scheme using either the conjugate gradient or the L-BFGS approach, but we choose to utilize GP regression also in the initial rotation phase. A similar initial phase where the lowest curvature mode is found by GP regression iterations is applied also in ref 20. With only a middle point and a randomized orientation given for the initial dimer, the GP-dimer algorithm proceeds as follows:

1. Evaluate accurate energy E_0 and force $\mathbf{F}_0$ at the middle point $\mathbf{R}_0$.
2. Check final convergence using accurate force $\mathbf{F}_0$.
3. Evaluate accurate energy E_1 and force $\mathbf{F}_1$ at $\mathbf{R}_1$.
4. Check rotational convergence using accurate forces $\mathbf{F}_0$ and $\mathbf{F}_1$.
5. Repeat initial rotations until rotational convergence:
   (a) Update the GP model based on the energy and force evaluations.
   (b) Rotate the dimer until rotational convergence using the GP approximation of the energy gradient.
   (c) Evaluate accurate energy E_1 and force $\mathbf{F}_1$ at $\mathbf{R}_1$.
   (d) Check rotational convergence using accurate forces $\mathbf{F}_0$ and $\mathbf{F}_1$.
6. Repeat GPR iterations until final convergence:
   (a) Update the GP model based on the energy and force evaluations.
   (b) Rotate and translate the dimer until early stopping or convergence using the GP approximation of the energy gradient.
   (c) Evaluate accurate energy E_0 and force $\mathbf{F}_0$ at $\mathbf{R}_0$.
   (d) Check final convergence using accurate force $\mathbf{F}_0$.

Figure 1 shows a two-dimensional illustration of the progression of the GP-dimer algorithm when finding a saddle point for dissociative adsorption of an H_2 molecule on a Cu(110) surface with fixed positions for the copper atoms. This example system is the same as the one used for testing the GP-NEB algorithm in ref 19. Each of the four graphs presents a cut of the GP approximation to the energy surface based on energy and force data evaluated at the points marked by + signs. The pink and red bars represent the dimer in the beginning and end of the initial rotation round or GPR iteration, respectively. Starting from an initial dimer coinciding with the two-dimensional cut of the coordinate space, the lowest curvature mode of the accurate energy surface is found after two initial rotation rounds, and the GP model extrapolates a saddle point close to the correct location already based on the four data points evaluated around the start point. After one more evaluation at the saddle point found on the approximate energy

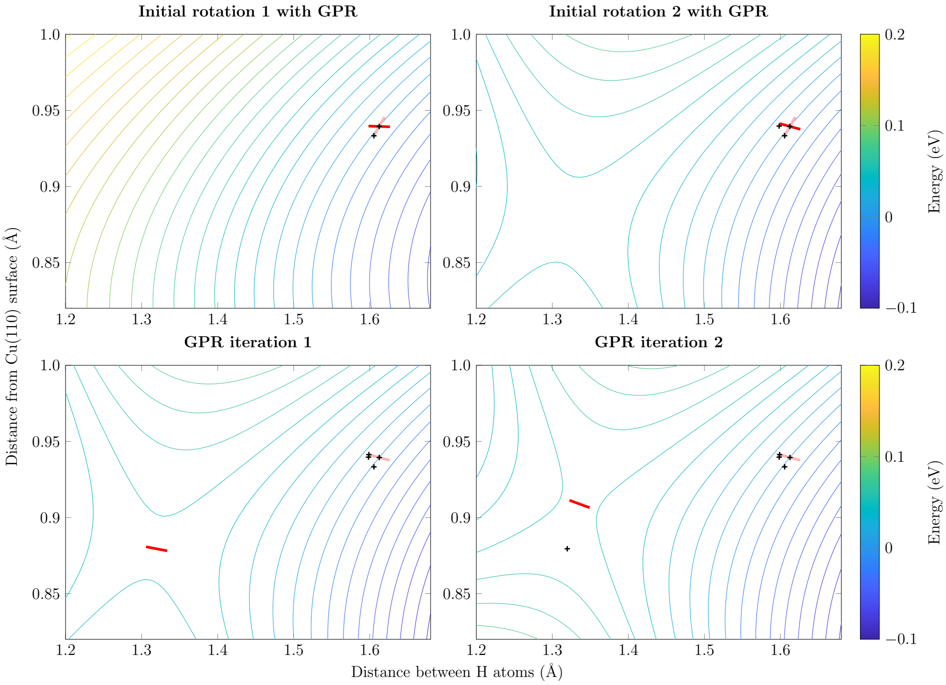


Figure 1: Two-dimensional cut through the energy surface of an H_2 molecule interacting with a Cu(110) surface. The H–H molecular axis lies in a plane parallel to the surface and perpendicular to the close-packed rows of Cu atoms. The upper graphs present GP approximations to the energy surface for the two initial rotation rounds and the lower graphs for the two GPR iterations of the GP-dimer algorithm. The training data points, marked with + signs, include both energy and force evaluations. The pink and red bars represent the dimer in the beginning and end of the initial rotation round or GPR iteration, respectively.

surface, the GP approximation is corrected and the middle point of the dimer converges to the correct saddle point.

Convergence of the GP-dimer algorithm has been reached when the maximum component of the accurate force $\mathbf{F}_0$ is below the final convergence threshold T_0 (here 0.01 eV/Å). Rotational convergence in the initial rotation phase is checked by calculating the preliminary rotational angle ω^*, given by eq 2, using the accurate forces $\mathbf{F}_0$ and $\mathbf{F}_1$. If more than one initial rotation round has been performed, also the angle between the converged orientations in the current and previous round is taken into account as an alternative criterion. The initial rotation phase is stopped when either of these angles is below T_ω (here 5°). The maximum number of initial rotation rounds is set to the number of degrees of freedom in the system.

During the initial rotation rounds, we use the rotation scheme of LBFGS-dimer with forces approximated by the GP model to find the lowest curvature mode on the approximate energy surface. As an exception to LBFGS-dimer, the new force $\mathbf{F}_1^{\text{new}}$ is not estimated with eq 19 after the rotation iterations but the GP approximation is used also there. A tighter convergence threshold $T_\omega^{\text{GP}} = \min\{0.01 \text{ rad}, T_\omega/10\}$ is used for both the preliminary rotational angle ω^* and the realized rotational angle ω, given by eq 5. Each initial rotation round is here started from the same initial orientation.

The LBFGS-dimer algorithm is used also for dimer relaxation in the actual GPR iterations where the dimer is both rotated and translated on the approximate energy surface. Again, the GP approximation of the new force $\mathbf{F}_1^{\text{new}}$ is used instead of estimating $\mathbf{F}_1^{\text{new}}$ with eq 19 or estimating the new curvature C^{new} with eq 11 after any rotation iteration. The rotational convergence threshold T_ω^{GP} for ω^* or ω is now set to 0.01 rad and the convergence threshold T_0^{GP} for the maximum component of $\mathbf{F}_0$ on the approximate energy surface is set to 1/10 of the lowest accurate maximum component of $\mathbf{F}_0$ evaluated so far. Here, dimer relaxation is always started from the same initial location with orientation obtained from the initial rotation phase.

In ref 19, the GP model based on inverse interatomic distances is coupled with an early stopping criterion constraining relative changes in the interatomic distances during the GP-NEB algorithm. The same early stopping criterion is used here for dimer relaxation inside the GPR iterations: After each translation iteration, there needs to exist an evaluated configuration $\mathbf{x}_{\text{eval}}$ so that

$$\forall i \in \mathrm{A}_{\mathrm{m}} \, \forall j \in \mathrm{A}_{\mathrm{m}} \cup \mathrm{A}_{\mathrm{f}} : \frac{2}{3} r_{i,j}(\mathbf{x}_{\text{eval}}) < r_{i,j}(\mathbf{R}_0) < \frac{3}{2} r_{i,j}(\mathbf{x}_{\text{eval}}). \tag{30}$$

If this condition does not hold, the last translation iteration is rejected and dimer relaxation stopped. Another early stopping criterion is based on the regular difference measure $\mathcal{D}_x$: After each translation iteration, there needs to exist an evaluated configuration $\mathbf{x}_{\text{eval}}$ so that

$$\mathcal{D}_x(\mathbf{R}_0, \mathbf{x}_{\text{eval}}) < L_x^{\text{es}}. \tag{31}$$

This criterion with $L_x^{\text{es}} = 0.5$ Å is applied also when using the stationary covariance functions k_x or $k_x^{\text{M}-5/2}$.

To guarantee that the early stopping criterion in eq 30 cannot be triggered by a single translation step taken from an evaluated data point, a following limitation rule is set for the step length of translation iterations inside the GPR iterations:[19] An individual atom $i \in \mathrm{A}_{\mathrm{m}}$ cannot move more than 99% of

$$\min_{j \in \mathrm{A}_{\mathrm{f}} \cup \mathrm{A}_{\mathrm{m}} \setminus \{i\}} r_{i,j}(\mathbf{R}_0)/6,$$

where the minimum is taken over all interatomic distances from that atom to any other atom in $\mathbf{R}_0$. If this limit is exceeded, the whole displacement vector is shortened so that the displacement of atom i is at the limit. A corresponding limitation rule to accompany the early stopping criterion in eq 31 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.

As in ref 19, an activation distance of 5 Å is applied here for the frozen atoms. This means that a frozen atom is taken into account in the covariance function of the GP model only if some of the moving atoms have been within the radius of 5 Å from the frozen atom in the configuration of the middle point of the dimer during the algorithm. The distances from the moving atoms to inactive frozen atoms are checked on each translation iteration inside the GPR iterations, and if new frozen atoms are activated, the GP model is updated.

4 Results

In this section, we present results for tests of the GP-dimer method with start points chosen randomly within the vicinity of saddle points for a dissociative adsorption of a hydrogen molecule on a Cu(110) surface and three different gas phase chemical reactions. The performance of the GP-dimer method with the inverse-distance covariance function $k_{1/r}$ and two stationary covariance functions, k_x and $k_x^{\text{M}-5/2}$, is reported in terms of the required number of energy and force evaluations and compared to two variants of the regular dimer method, described above as CG-dimer and LBFGS-dimer. In addition, we compare the number of evaluations required for finding the lowest curvature mode of the energy surface when performing initial rotations using the Gaussian process regression, conjugate gradient, or L-BFGS approach.

4.1 Application to H_2 dissociation on Cu(110)

Our first example transition is a dissociative adsorption of an H_2 molecule on the Cu(110) surface. The same transition has been used in ref 19 for testing the GP-NEB algorithm for finding the minimum energy path between given initial and final states. In the test system here, the two H atoms are allowed to move, whereas the Cu atoms are frozen. The energy surface is described in ref 1. The start point of the algorithm is chosen by a random displacement of 0.02, 0.05, 0.1, 0.2, 0.3, 0.4, 0.6, 1, 2, or 3 Å from the saddle point of the example transition in the six-dimensional coordinate space. Ten different start points and randomly chosen initial orientations are used for each distance. Figure 1 illustrates the progression of the GP-dimer algorithm in an easy example where the start point and the initial orientation coincide with the same two-dimensional cut of the coordinate space as the saddle point.

Figure 2 shows the number of energy and force evaluations required for convergence to a saddle point with the GP-dimer method and the two variants of the regular dimer method, CG-dimer (orange) and LBFGS-dimer (blue). In addition to the inverse-distance covariance function $k_{1/r}$ (green), GP-dimer results are shown also for the squared exponential covariance function k_x (red) and Matérn-5/2 covariance function $k_x^{\mathrm{M}-5/2}$ (violet). In almost all cases, GP-dimer requires fewer evaluations than the regular dimer methods, and the difference increases when moving the start point farther away from the saddle point of the example transition. With start points closer than 0.5 Å to the saddle point, there are only small differences in the performance between the three covariance functions in the GP-dimer calculations, but the benefits of the inverse-distance covariance function become visible with larger distances. Figure 3 shows an example of the behaviour of the GP-dimer method with different covariance

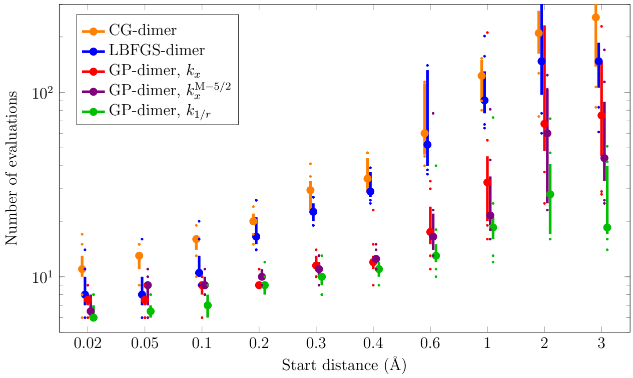


Figure 2: Number of energy and force evaluations required for convergence to a saddle point in the H_2/Cu(110) example using the regular CG-dimer (red) and LBFGS-dimer (blue) methods and the GP-dimer method with the squared exponential (red), Matérn-5/2 (violet), and inverse-distance (green) covariance functions. The distance of the start point of the calculation from the example saddle point is shown on the horizontal axis, and the vertical axis represents the number of evaluations in logarithmic scale. The large dots present the median number of evaluations among 10 randomly chosen start positions. The bars present the interval between the third and eighth largest numbers, and the two smallest and largest numbers are presented by small dots if not included in the interval.

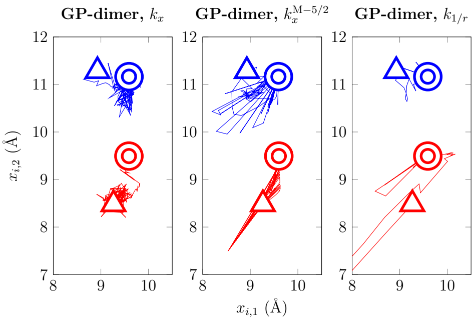


Figure 3: Example of the movement of the two H atoms (blue and red) during the GP-dimer algorithm in the H_2/Cu(110) example when using the squared exponential (left), Matérn-5/2 (middle), and inverse-distance (right) covariance functions. The locations of the atoms are shown as projections on a plane parallel to the Cu(110) surface. The triangles represent the start configuration, and the double circles represent the saddle point where the algorithm converges.

functions with a start distance of 3 Å. The blue and red lines present the movement of the two H atoms projected on the plane of the Cu(110) surface during the algorithm. When the inverse-distance covariance function is used (right), convergence is reached after 40 evaluations. With Matérn-5/2 (middle) and squared exponential (left) covariance functions, convergence to the same saddle point requires 170 and 228 evaluations, respectively.

4.2 Application to chemical reactions

Another set of test examples studied here involves three chemical reactions ($N_\mathrm{m} \leq 14$). The electronic structure computations for the energy and atomic forces are performed using the PM3 semi-empirical approach[31] as implemented within the ORCA suite of programs.[32] Reaction 1 is a simple addition of N_2O and ethylene to form oxadiazole, reaction 2 is the rearrangement of allyl vinyl ether to form 1-pentene-5-one, and reaction 3 is the removal of sulfur dioxide from butadiene sulfone.[33] The activation energies for the example reactions are relatively high: 1.86 eV, 3.36 eV, and 2.41 eV, respectively. For each of the reactions, the saddle point used as the center of start points is confirmed to have a single negative eigenvalue of the Hessian. The reactant, saddle point, and product state configurations of the example reactions are illustrated in Figure 4 alongside the corresponding result graphs.

Start points for the dimer calculations are chosen by a random displacement of 0.02, 0.05, 0.1, 0.2, 0.3, 0.4, 0.6, or 1 Å from the example saddle point in the $3N_\mathrm{m}$-dimensional coordinate space. Ten different start points and randomly chosen initial orientations are again used for each value of the distance. As shown in Figure 4, the pattern of the results is quite similar for each of the three test examples. Unlike in the H_2/Cu(110) example, the LBFGS-dimer method performs here clearly better than CG-dimer. The three variants of the GP-dimer method require again significantly fewer evaluations than the regular dimer methods, but the difference between the stationary and inverse-distance covariance functions starts to become appreciable already at 0.1 Å. From some start positions, the algorithms may end up in configurations where the energy and force evaluations fail. In such cases, the number of required evaluations is considered to be above 300.

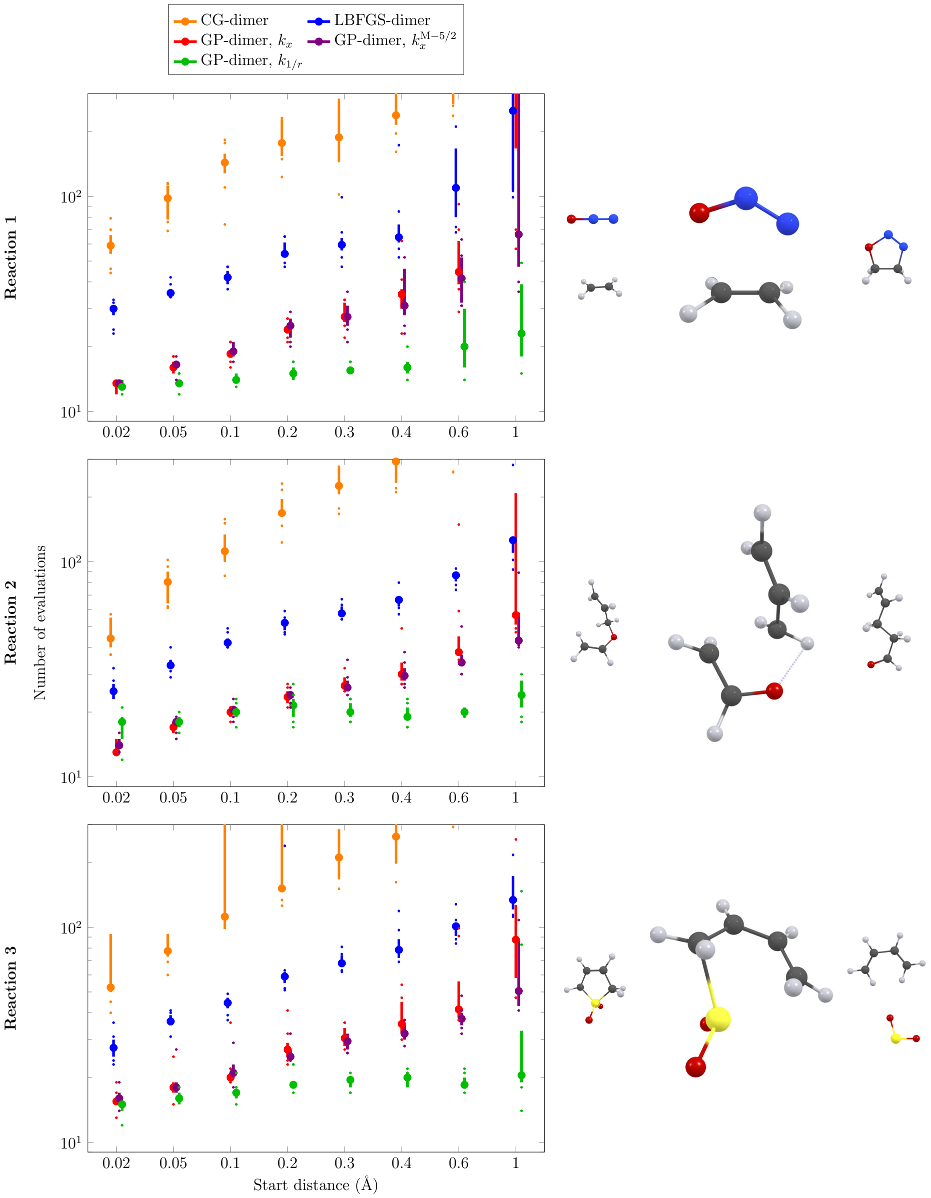


Figure 4: Number of energy and force evaluations required for convergence to a saddle point in three test examples using the regular CG-dimer (red) and LBFGS-dimer (blue) methods and the GP-dimer method with the squared exponential (red), Matérn-5/2 (violet), and inverse-distance (green) covariance functions. The distance of the start point of the calculation from the saddle point of the example reaction is shown on the horizontal axis, and the vertical axis represents the number of evaluations in logarithmic scale. The large dots represent the median number of evaluations among 10 randomly chosen start positions. The bars present the interval between the third and eighth largest numbers, and the two smallest and largest numbers are represented by small dots if not included in the interval. The reactant, saddle point and product state configurations for each example reaction are visualized with the following atom colors: C, dark gray; H, light gray; O, red; N, blue; S, yellow.

4.3 Initial rotations

In the GP-dimer method, the initial rotation phase is performed using a Gaussian process regression approach where the lowest curvature mode at the start point is found by rotating the dimer on the approximate energy surface and refining the GP model based on accurate evaluations. To demonstrate the savings as compared to the conjugate gradient or L-BFGS approaches, we present results from separate tests where the rotations at the start point are continued until the preliminary rotation angle ω^*, given by eq 2, is below 5°. If ω^* is based on estimated forces, the rotational convergence is confirmed by evaluating the accurate forces and the rotations are continued if necessary.

Figure 5 presents the number of evaluations required for rotational convergence in the H_2/Cu(110) example with the conjugate gradient (red), L-BFGS (blue), and GP regression approaches (red, violet, and green for squared exponential, Matérn-5/2, and inverse-distance covariance functions, respectively). For the GP regression approach, the median of the number of evaluations remains between four and six with any of the three covariance functions. The median for the L-BFGS approach is 1–3 evaluations and the median for the conjugate gradient approach 3–11 evaluations larger than for the GP regression approach, but those results include more outliers. Due to the local nature of the training data set in the initial rotation phase, there are only small differences between the stationary and inverse-distance covariance functions.

Figure 6 presents corresponding results for the three chemical reaction examples. Again, the GP regression approach consistently converges with fewer evaluations than the comparison methods, although the difference from the L-BFGS approach is small, especially when using the stationary covariance functions. The performance of the conjugate gradient approach is significantly worse.

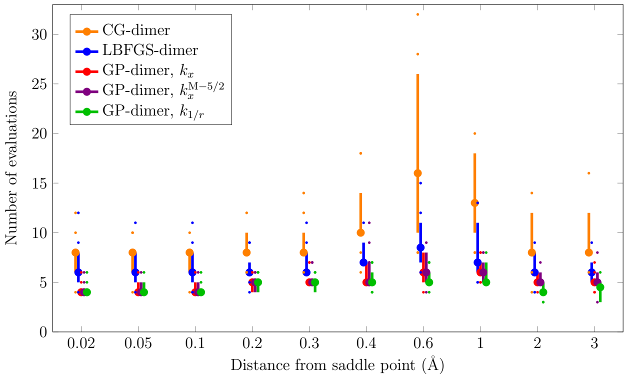


Figure 5: Number of energy and force evaluations required for rotations to the lowest curvature mode of energy in the H_2/Cu(110) example with the conjugate gradient (red) and L-BFGS (blue) approaches and with the Gaussian process regression approach using the squared exponential (red), Matérn-5/2 (violet), and inverse-distance (green) covariance functions. The distance between the locations of the middle point of the dimer and the saddle point of the example transition is shown on the horizontal axis. The large dots present the median number of evaluations among 10 randomly chosen start positions. The bars present the interval between the third and eighth largest numbers, and the two smallest and largest numbers are presented by small dots if not included in the interval.

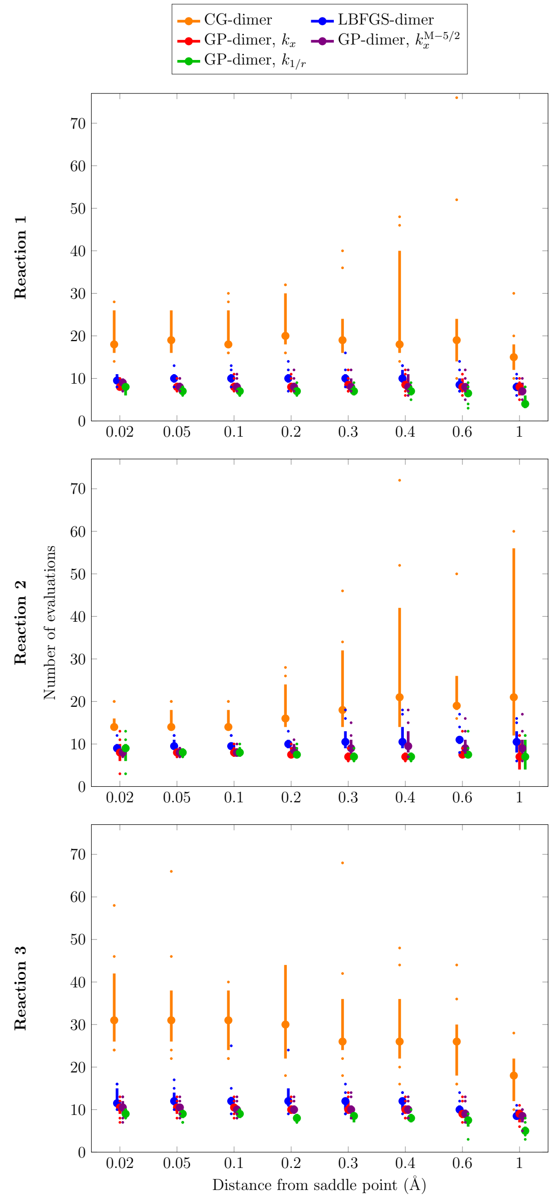


Figure 6: Number of energy and force evaluations required for rotations to the lowest curvature mode of energy in the three chemical reaction examples with the conjugate gradient (red) and L-BFGS (blue) approaches and with the Gaussian process regression approach using the squared exponential (red), Matérn-5/2 (violet), and inverse-distance (green) covariance functions. The distance between the locations of the middle point of the dimer and the saddle point of the example reaction is shown on the horizontal axis. The large dots present the median number of evaluations among 10 randomly chosen start positions. The bars present the interval between the third and eighth largest numbers, and the two smallest and largest numbers are presented by small dots if not included in the interval.

5 Discussion

The results presented here show that the inverse-distance covariance function suggested for GP-NEB calculations in ref 19 is beneficial also when applying the Gaussian process regression approach to minimum mode following calculations. The improved covariance function and the accompanying early stopping criterion are especially important when the start point for the calculation is not close to a saddle point. Already when the start point is displaced 0.1 Å from a saddle point, the GP-dimer method with the inverse-distance covariance function may require a significantly smaller number of energy and force evaluations to reach convergence than when using a stationary covariance function. Our results show also generally that the GP regression approach (no matter which covariance function is used) gives an advantage also when starting really close to the saddle point compared to the usual implementations of the dimer method based on conjugate gradient or L-BFGS algorithms. This applies to the initial rotations as well as to the translations of the dimer in the climb up the energy surface. Even though stationary covariance functions give convergence in the examples presented here, similar problems can arise as seen in GP-NEB calculations where the inclusion of large atomic forces can lead to failure in calculations based on stationary covariance functions.[19] Ultimately, the GP-dimer method can be used in repeated saddle point searches starting from a given local minimum to map out relevant low-lying saddle points in long time scale simulations.[3,4] We expect the robustness of the inverse-distance covariance approach will then be even more important.

We have assumed here that no information about the energy surface is available in the beginning of the minimum mode following calculation, only the coordinates and a random orientation of the dimer. If other information is available, the initial rotation phase may be unnecessary. This is the case, for example, if the start point has been obtained from an NEB calculation on some approximate energy surface, based for example on a lower level of electronic structure theory, or if an NEB calculation has converged only to a large tolerance. In these cases, information from the NEB calculation can be utilized when training the GP model, and that can make the GP regression approach even more useful.

After finding a saddle point, all eigenvalues of the Hessian at the saddle point are typically required to estimate the transition rate using the harmonic approximation to transition state theory. This can involve substantial computational effort for large systems. The GP model learned during the minimum mode following calculation gives a probability distribution for the energy surface, which can be used to estimate the Hessian and its eigenvalues as well as the uncertainty of these estimates and the calculated transition rate. If the rate cannot be estimated reliably enough, new energy and force calculations can be performed in a systematic way to update the GP model until the required confidence levels have been reached. Uncertainties of the second derivatives could also be utilized for deciding when to evaluate also image 1 (once or even repeatedly until rotational convergence) during the algorithm, which may become beneficial when starting far from a saddle point.

Acknowledgements

This work was financially supported by the Academy of Finland (Grant 278260; Flagship programme: Finnish Center for Artificial Intelligence, FCAI, Grants 320181, 320182, 320183) and the Icelandic Research Fund. O.-P.K. was supported by the Finnish Cultural Foundation (Grant 180536), and V.Á. was supported by a doctoral fellowship from the University of Iceland Research Fund. 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] Henkelman, G.; Jónsson, H. Long time scale kinetic Monte Carlo simulations without lattice approximation and predefined event table. *J. Chem. Phys.* **2001**, 115, 9657.

[4] Plasencia Gutiérrez, M.; Argáez, C.; Jónsson, H. Improved minimum mode following method for finding first order saddle points. *J. Chem. Theory Comput.* **2017**, 13, 125.

[5] Ásgeirsson, V.; Jónsson, H. Exploring potential energy surfaces with saddle point searches. In *Handbook of Materials Modeling: Methods: Theory and Modeling*; Andreoni, W., Yip, S., Eds.; Springer: Cham, Switzerland, 2018.

[6] Henkelman, G.; Jónsson, H. A dimer method for finding saddle points on high dimensional potential surfaces using only first derivatives. *J. Chem. Phys.* **1999**, 111, 7010.

[7] Munro, L. J.; Wales, D. J. Defect migration in crystalline silicon. *Phys. Rev. B* **1999**, 59, 3969.

[8] Malek, R.; Mousseau, N. Dynamics of Lennard-Jones clusters: a characterization of the activation-relaxation technique. *Phys. Rev. E* **2000**, 62, 7723.

[9] Olsen, R. A.; Kroes, G. J.; Henkelman, G.; Arnaldsson, A.; Jónsson, H. Comparison of methods for finding saddle points without knowledge of the final states. *J. Chem. Phys.* **2004**, 121, 9776.

[10] Heyden, A.; Bell, A. T.; Keil, F. J. Efficient methods for finding transition states in chemical reactions: comparison of improved dimer method and partitioned ration function optimization method. *J. Chem. Phys.* **2005**, 123, 224101.

[11] Kästner, J.; Sherwood, P. Superlinearly converging dimer method for transition state search. *J. Chem. Phys.* **2008**, 128, 14106.

[12] Voter, A. F. Hyperdynamics: accelerated molecular dynamics of infrequent events. *Phys. Rev. Lett.* **1997**, 78, 3908.

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

[14] 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.

[15] 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.

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

[17] 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 is available as e-print arXiv:1703.10423.

[18] 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.

[19] Koistinen, O.-P.; Ásgeirsson, V.; Vehtari, A.; Jónsson, H. Nudged elastic band calculations accelerated with Gaussian process regression based on inverse interatomic distances. *J. Chem. Theory Comput.* **2019**, doi:10.1021/acs.jctc.9b00692.

[20] Denzel, A.; Kästner, J. Gaussian process regression for transition state search. *J. Chem. Theory Comput.* **2018**, 14, 5777.

[21] Fletcher, R.; Reeves, C. M. Function minimization by conjugate gradients. *Comput. J.* **1964**, 7, 149.

[22] Polak, E.; Ribière, G. Note sur la convergence de méthodes de directions conjuguées. *ESAIM: Mathematical Modelling and Numerical Analysis - Modélisation Mathématique et Analyse Numérique* **1969**, 3, 35.

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

[24] Liu, D. C.; Nocedal, J. On the limited memory BFGS method for large scale optimization. *Math. Program.* **1989**, 45, 503.

[25] 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.

[26] 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.

[27] 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, MA, 2003; pp 1057–1064.

[28] 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.

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

[30] 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.

[31] Stewart, J. J. P. Optimization of parameters for semiempirical methods I. Method. *J. Comput. Chem.* **1989**, 10, 209.

[32] Neese, F. Software update: the ORCA program system, version 4.0. *Wiley Interdiscip. Rev. Comput. Mol. Sci.* **2018**, 8, e1327.

[33] Birkholz, A. B.; Schlegel, H. B. Using bonding to guide transition state optimization. *J. Comput. Chem.* **2015**, 36, 1157.

Aalto-DD 222/2019

SIGILLUM UNIVERSITATIS ISLANDIAE

9 789526 088501

ISBN 978-952-60-8850-1 (printed)
ISBN 978-952-60-8851-8 (pdf)
ISSN 1799-4934 (printed)
ISSN 1799-4942 (pdf)

Aalto University
School of Science
Department of Computer Science
www.aalto.fi