# TRANSITION-STATE OPTIMIZATION METHODS USING INTERNAL COORDINATES

## **New Transition-State Optimization Methods**

By

### **Carefully Selecting Appropriate Internal Coordinates**

By

Sandra Rabi

H.B.Sc. (Chemistry)

A Thesis

Submitted to the School of Graduate Studies

in Partial Fulfillment of the Requirements

for the Degree

Doctor of Philosophy

McMaster University

©Copyright by Sandra Rabi

DOCTOR OF PHILOSOPHY (2014) (Chemistry) McMaster University Hamilton, Ontario. Canada

TITLE: New Transition-State Optimization Methods By Carefully Selecting Appropriate Internal Coordinates

AUTHOR: Sandra Rabi H.B.Sc. - Honours Chemistry (Physical and Analytical Specialization) (McMaster University, Hamilton, Ontario. Canada)

SUPERVISOR: Dr. Paul W. Ayers

NUMBER OF PAGES: xxvi, 263

### ABSTRACT

Geometry optimization is a key step in the computational modeling of chemical reactions because one cannot model a chemical reaction without first accurately determining the molecular structure, and electronic energy, of the reactants and products, along with the transition state that connects them. These structures are stationary points—the reactant and product structures are local minima, and the transition state is a saddle point with one negative-curvature direction—on the molecular potential energy surface. Over the years, many methods for locating these stationary points have been developed. In general, the problem of finding reactant and product structures is relatively straightforward, and reliable methods exist. Converging to transition states is much more challenging.

Because of the difficulty of transition-state optimization, researchers have designed optimization methods specifically for this problem. These methods try to make good choices for the initial geometry, the system of coordinates used to represent the molecule, the initial Hessian, the Hessian updating method, and the step-size. The transition-state optimization method developed in this thesis required considering all of these methods. Specifically, a new method for finding an initial guess geometry was developed in chapter 2; good choices for a coordinate system for representing the molecule were explored in chapters 2 and 6; different choices for the initial Hessian are considered in chapter 5; chapters 3 and 4 present, and test, a sophisticated new method for updating the Hessian and controlling the step-size during the optimization. The methods created in the process of this research led to the development of *Saddle*, a general-purpose geometry optimizer for transition states and stable structures, with and without constraints on the molecular coordinates. *Saddle* can be run in conjunction with the *Gaussian* program or almost any other quantum chemistry program, and it converges significantly more often than the other traditional methods we tested.

I dedicate this dissertation work to my loving parents, to my father who has been my main source of inspiration, and to my mother who showed me that love has no boundaries. I am grateful for your encouragement and support throughout this journey. I also dedicate this work to my sisters, Nataly and Santa, for always being there for me.

### ACKNOWLEDGEMENTS

I would like to start off by thanking my wonderful supervisor Dr. Paul Ayers. I don't think there are words that can describe my appreciation for all of the tremendous help that I have received from him. Thank you Paul for being a genius, I don't think I could have done anything in this thesis without your brilliant mind. Thank you for being a great mentor, and for being so generous and patient. I am grateful for having a lifelong friend who I know will always be there for me. A million thank you Paul for being so awesome.

I would also like to thank my committee members, Dr Randall Dumont, Dr Paul Berti and Dr. Alex Bain. Their guidance, patience and comments were always helpful for my project.

A big thank you goes to Dr. Toon Verstraelen for all of the help that he provided for my thesis, especially during my visit to Belgium. Toon, I thought I would never learn programming, but you proved me wrong. Thank you for being a great mentor and a good friend.

I would also like to thank Dr. Steven Burger for all of the helpful guidance during my first year of my journey. Steven helped me have a smooth transition from being an undergraduate to being a new researcher. Thanks Steven for all your patience.

A huge thank you goes to Santa Rabi and Christopher Haddad. Their great help with projects during my last year at McMaster was very much appreciated. It was a very stressful and busy time for me, but they were patient enough for my bossy attitude, and managed to make me laugh with their jokes.

I would also like to give thanks to all of the Ayers group members that I have met throughout my years there. Their friendly attitude made my experience more meaningful. Particularly, a big giant thank you goes to Rogelio Cuevas, for being the big brother that I can always depend on. Thank you Rogelio for all of the great memories that you shared with the Rabi sisters. I am truly blessed to have a great friend like you.

Of course I would have never reached this level without my great family. Thank you dad and mom for all of the sacrifices that you have made for my sisters and me. Thank you Nataly and Santa for being my best friends and my sisters. I love you all!

## **Table of Contents**

Chapte	er 1 Introduction 1-55
1.1	Introduction
1.2	The Molecular Potential Energy Surface (PES)4
A.	The Born-Oppenheimer Approximation4
B.	Characterization of Potential Energy Surfaces7
C.	Numerical Calculations Using Potential Energy Surfaces
1.3	Coordinate Systems for Expressing the Potential Energy Surface (PES) 12
A.	Cartesian Coordinate System 12
B.	Systems of Redundant and Nonredundant Internal Coordinates
C.	Transforming Between Internal and Cartesian Coordinate Systems15
D.	The Problem of Linear Dihedrals 19
1.4	Numerical Methods for Optimization
A.	Newton's Method
B.	Quasi-Newton Methods
C.	Step-Size Control
1.	Line search method
2.	Trust radius methods
1.5	The Initial Geometry Guess for Transition State Optimization

А.	The Problem of Finding Good Initial Guesses
В.	Single-Ended Methods
C.	Double-Ended Methods
1.6	Summary
1.7 R	eferences
Chapt	er 2: Generating Initial Guesses for Transition-State
	Optimization by Interpolating in Robust Redundant
	Internal Coordinates 56-114
2.1	Overview
2.2	Motivation
2.3	Background
A.	Methods for Guessing Transition States
В.	Redundant Internal Coordinates61
C.	Preview of Our Results
2.4	Methods
A.	Redundant Internal Coordinates
В.	Choosing Robust Redundant Internal Coordinates
1	. Interatomic Distances
2	2. Angles between Bonds

3	5. Dihedral Angles; Rotations Around Bonds
C.	Converting from Redundant Internal Coordinates to Cartesian Coordinates 71
D.	An Inexpensive Double-Ended Transition-State Guesser
2.5	Results
A.	Testing Protocol
В.	Overview of Results from the Transition-State Guessing Methods77
2.6	Summary
App	endix 2.1: Proof that pmax = 0.5 for Method 2 and Method 3
App	endix 2.2: Reaction Database
2.7	References 107
Chapt	er 3 A Reliable Method for Transition-State Optimization
	Using Partial Computation of the Hessian and Robust
	Redundant Internal Coordinates115-174
3.1	Overview
3.2	Introduction 116
3.3	Key Components of the <i>Saddle</i> Algorithm 120
A.	Initialization

C. Coordinate transformation between Cartesian and redundant internal coordinates
D. On-the-fly addition of new internal coordinates 128
E. "Reduced" coordinates to describe key chemical changes 129
F. The V matrix of delocalized reduced and internal coordinates 130
G. Updating procedure for the Hessian matrix: The secant condition in v-
coordinates 135
H. Updating procedure for the Hessian matrix: Finite differences
I. Updating procedure for the Hessian matrix: Quasi-Newton methods 137
J. Updating procedure for the Hessian matrix: Eigenstructure Modification 141
K. Step direction: Scaled-Newton or rational function step? 143
1. Trust-region image potential (TRIM) (Scaled Newton step) 144
2. Rational function optimization (RFO) 145
L. Step-size determination: Trust radius methods
1. Energy-based trust radius updating 147
2. Gradient-based trust radius updating 148
3. Summary 150
M. Convergence Criteria 151
3.4 Overview of <i>Saddle</i> 151

3.5	Numerical Tests 154
A.	Computational Methods 154
В.	Assessment: Methods for step direction (TRIM, RFO) and step size (trust radius)
C.	Assessment: Quasi-Newton methods 155
D.	Assessment: Comparison to the <i>Berny</i> Optimizer
3.6	Summary 160
Appe	endix 3.1. The secant condition in delocalized nonredundant internal coordinates.
Appo	endix 3.2. The set of reactions used for testing
3.7	References 169
Chapt	er 4 Systematic Assessment of Transition-State Optimization
	Methods Using Randon Transition-State Guesses for a
	Database of 131 Reactions175-200
4.1	Statement of the Problem 176
4.2	Introduction 176
4.3	Testing Protocol 179
А.	A Database of Chemical Reactions for Testing Transition-State Optimizers

A Systematic Method for Generating Initial Guesses of Decreasing Quality. 180
A Criterion for Determining Whether a Method Converges to the Correct
Insition State
Results: Comparing the Performance of Transition-State Optimizers
Summary 193
References
er 5 Systematic Assessment of Quantum-Chemical Initial
Hessian Approximation for Transition-State
Optimization201-212
Overview
Introduction
Testing Procedure 204
Discussion 206
References
er 6 Automatic Detection of Key Chemical Coordinates for
Reeduced-Coordinate Approaches to Transition-State
<b>Optimization</b> 213-227
Overview
Introduction

6.3	Protocol for Automatically Selecting Reduced Coordinates 216
6.4	Testing Protocol 216
6.5	Results and Discussion 222
6.6	References 226
Chapt	er 7 Conclusions & Future Work228-263
7.1	Summary
7.2	Future Work and Other Projects Not Included in the Thesis
A.	Overview
B.	Extensions and Further Development of the Topics Covered in the Thesis 233
1	. Improved Initial Guesses for Transition-State Geometries
2	2. Alternative Secant Conditions for <i>Saddle</i>
3	8. Strict Separation of Reduced and Nonredundant Coordinates 234
4	Nonredundant Alternatives to Delocalized Internal Coordinates
5	5. Redundant Alternatives to Delocalized Internal Coordinates
6	5. Finding the best Quasi-Newton Hessian
7	7. Testing Robust Dihedral Coordinates
8	8. Optimizing Program Parameters
ç	Extensions: Minimization and Frozen Coordinates
C.	Topics Not Covered in the Thesis

	1.	Finding the Reaction Path with Sequential Quadratic Programming	246
	2.	Study of the Epoxide Hydrolase Enzyme	250
	3.	Computational & Theoretical Studies of Chemical Reactivity	254
7.3		References	256

### **List of Figures**

Chapter 1	Introduction 1-55		
Figure 1.1	A four-well two-dimensional potential energy surface, showing four		
	minima and four saddle points		
Chapter 2:	Generating Initial Guesses for Transition-State		
	Optimization by Interpolating in Robust Redundant		
	Internal Coorinates 56-114		
Figure 2.1	Representation and notational convention for a dihedral angle. Notice that		

- Figure 2.3 A performance plot showing the percentage of reactions (y axis) that had converged within a given number iterations (x axis) for initial transitionstate guesses obtained by the three different methods when  $p_{post}$

(Eq.	(2.20)) is chosen		83	3
------	-------------------	--	----	---

## 

# 

### **List of Tables**

Chapter 2:	Generating	Initial	Guesses	for	Trans	sition-State
	Optimization	by In	terpolating	in	Robust	Redundnt
	Internal Cool	rinates .	•••••	•••••	•••••	56-114

Table 2.1.	Assessment of the overall performance of the three different methods of
	guessing transition states

# Chapter 3 A Reliable Method for Transition-State Optimization Using Partial Computation of the Hessian and Robust Redundant Internal Coordinates.......115-174

Table 3.1 The number of gradient evaluations required to converge the reactions in Appendix 3.2 using the HF/3-21G(d) method for different methods for computing the step-direction (cf. section 3.3.K) and controlling the step size with a trust radius (cf. section 3.3.L). The Bofill-1994 quasi-Newton update, with condition (3.55) imposed, was used. The average number of gradient evaluations when the method converges to the correct transition state is tabulated so that the computational cost of different methods can be compared. For each reaction, the result for the most efficient version(s) of

our approaches is bold-faced. When the Berny transition-state optimization is more efficient than any of our approaches, its results are underlined......

Appendix 3.2 The set of reactions used for testing ......165

## 

Table 4.1 The number of steps, averaged over all transition-state guesses of a given quality for all 131 reactions, required for various methods to converge to the targeted transition state. The initial guesses are constructed by randomly perturbing all the Cartesian coordinates of the correct transitionstate structure by a specified amount,  $\varepsilon$ , as expressed in Eqs. (4.1) and (4.2). If, for a given initial guess, a computational method fails (crashes), does not converge within 200 iterations, or converges to the wrong transition state, then that calculation is not included in the average ......186

- Table 4.3 The number of steps, averaged over all transition-state guesses of a given quality for all 131 reactions, required for various methods to converge to the targeted transition state. The initial guesses are constructed by randomly perturbing the key chemical coordinates of correct transition-state structure so that the deformed structure differs from the true transition-state structure by a specified amount, ε. (Cf. Eqs. (4.3) and (4.4).) If, for a given initial guess, a computational method fails (crashes), does not converge within 200 iterations, or converges to the wrong transition state, then that calculation is not included in the average.....190

# 

- Table 6.2Percentage convergence for the various methods for choosing the reduced-<br/>coordinates (user-specified, automatically generated, or both (the union<br/>of the two sets) and the redundant internal coordinates (using only the TS<br/>(transition-state structure), or using all available structures (the transition-<br/>state structure, the reactant structure, and the product structure)).......221

#### PREFACE

This thesis descibes an optimization method that was developed by the author, together with her supervisor (Prof. Paul Ayers) and their collaborator (Dr. Toon Verstraelen). This project grew from the author's frustration with existing techniques for optimizing transition state structures in chemical reactions. The method she developed applies to additional problem (finding local minima on the molecular potential energy surface, constrained geometry optimization, etc.) but the emphasis of the method, and of this thesis, is on optimizing transition states. As part of the thesis, the author developed a program called *Saddle*, which can be used to do almost all of the types of geometry optimization that are relevant in chemistry. *Saddle* will eventually be incorporated in the *Horton* program package, but is currently stand-alone software.

All of the chapters in the thesis were written by the author and edited by Prof. Ayers; Prof. Verstraelen also made some minor suggestions for elaboration and revision of the document. The first chapter of the thesis provides background, and the next three chapters present the key components of, and results for, the transition-state optimizer developed by the author. Chapters 5 and 6 present minor variations of the method, showing how the method can be simplified and improved by adjusting the initial guess. Chapter 7 provides perspective on the results presented in the thesis.

Chapter 1 is a detailed introduction and literature review about geometry optimization, focusing on transition state optimization. The key features of the problem of optimizing transition states are discussed, and existing approaches are reviewed. This has been submitted as an invited review article in the *Journal of Theoretical and* 

#### Computational Chemistry.

Chapter 2 presents an extremely efficient and highly effective method for approximating the reaction path between a chemical reactant and its product, and estimating the transition-state structure along that path. This method is extremely fast because it does not require any evaluations of the potential energy surface: it is based on interpolating between the reactant and product structures, in redundant internal coordinates. Despite its conceptual simplicity and computational efficiency, the estimate of the transition-state structure is quite good in almost all cases. The new method was programmed by the author, using some utility-routines from Prof. Verstraelen. After initial testing by the author, she supervised two undergraduate students (Miss Santa Rabi and Mr. Christopher Haddad) as they performed systematic tests. The resulting paper has been submitted to the *Journal of Chemical Theory and Computation*.

Chapter 3 describes the transition state optimization method that was developed. It starts with an idea proposed by Dr. Steven Burger and Prof. Ayers; they recommended refining the molecular Hessian using finite-differences in certain key chemical coordinates. (These are typically the internuclear distances associated with the bonds that break and/or form in the chemical reaction of interest.) The mathematics of the method was worked out by the author with the assistance of Prof. Ayers. Prof. Verstraelen then helped guide the programming of the method, and Prof. Ayers helped the author debug the program and assisted in refining the algorithm. The resulting paper has been submitted to the *Journal of Chemical Theory and Computation*.

Chapter 4 presents a systematic study of the performance of the method in chapter

3, as compared to the conventional approaches available in the *Gaussian* program. The author built the testing database and ran all the calculations. The resulting paper has been submitted to the *Journal of Chemical Theory and Computation*.

Chapter 5 shows how different choices for the initial Hessian change the performance of transition-state optimizers. Somewhat remarkably, one does not need to expend much effort in evaluating the initial Hessian: a calculation with a minimal basis set suffices. The idea for this chapter came from the author and Prof. Ayers; the author supervised Miss Rabi and Mr. Haddad as they performed the systematic computational tests for this chapter. The resulting paper has been submitted to the *Journal of Chemical Theory and Computation*.

Chapter 6 shows that it is not mandatory for the user to explicitly choose the key chemical coordinates in the method of chapter 3: the coordinates can be identified automatically. Different ways of automatically selecting the coordinates are considered, and their relative efficiency is compared. The methods explored were programmed by the author, and tested by Miss Rabi and Mr. Haddad, under the author's supervision. The resulting paper is in preparation for *Chemical Physics Letters*.

A summary of the content of the thesis, along with directions for further research, is provided in Chapter 7. This chapter also overviews research projects that the author was involved in, but which do not appear in the thesis.

**Chapter 1** 

# Introduction

### **1.1 Introduction**

Locating stationary points on the molecular potential energy surface that lie along the chemical reaction path is the first step in modeling chemical reactions computationally. Geometry optimization methods locate these points. For example, the endpoints of the chemical reaction path are the reactant and product complexes; these structures are minima on the molecular potential energy surface. If there are additional minima on the reaction path, those structures correspond to (meta)stable reactive intermediates. Maxima on the chemical reaction path (saddle points on the molecular potential energy surface) are transition states.<sup>1,2</sup>

Once these stable structures (minima) and the transition states linking them have been determined, one can estimate many kinetic and thermodynamic properties, including rate constants (from the height of reaction barriers, using transition-state theory) and equilibrium constants (from the relative depth of the minima).<sup>3</sup> In many multi-step reactions, only the reactant and the final product are observed experimentally. In these cases, knowing the minimum energy path of a reaction provides detailed information about the reaction mechanism, including all the different intermediates and transition state structures. This knowledge can be used to design catalysts, leading to more efficient and environmentally friendly synthetic procedures.

Many methods have been developed for geometry optimization. The efficiency of these methods depends on several factors: the choice of coordinates, the initial geometry, the initial Hessian (second derivative of the energy), the choice of step size in the iterative procedure, and the choice of the Hessian updating methods.<sup>4-9</sup> It is generally easier to find

a minimum structure than it is to optimize to a transition state because in minimization, a structure with lower energy is always preferable. In transition-state finding, the total energy may either increase or decrease when approaching the saddle point.

The difficulty of transition-state optimization has led to specialized methods for providing good initial guesses of transition state structures and for optimizing transition state structures. In this chapter, an overview of recent geometry optimization methods is provided, focusing on transition-state finding. For completeness, and to establish our notation, we quickly review the definition and properties of the molecular potential energy surface. We then compare the different choices of coordinates, internal and Cartesian, that can be used to express the potential energy surface, discussing their advantages and disadvantages for optimization. Quasi-Newton and step-size control methods are also described in detail, with special emphasis on their application to transition-state optimization. Finally, methods for finding an accurate initial approximation to the transition state structure are reviewed.

### **1.2** The Molecular Potential Energy Surface (PES)

#### A. The Born-Oppenheimer Approximation

The energy operator (Hamiltonian) for a molecule is:<sup>10,11</sup>

$$\hat{H} = -\sum_{\alpha} \frac{\hbar^2}{2M_{\alpha}} \nabla_{\alpha}^2 - \sum_{i} \frac{\hbar^2}{2m_e} \nabla_i^2 + \sum_{\alpha} \sum_{\beta > \alpha} \frac{Z_{\alpha} Z_{\beta} e^2}{4\pi \varepsilon_0 R_{\alpha\beta}} - \sum_{\alpha} \sum_{i} \frac{Z_{\alpha} e^2}{4\pi \varepsilon_0 r_{i\alpha}} + \sum_{j} \sum_{i>j} \frac{e^2}{4\pi \varepsilon_0 r_{ij}}$$

$$(1.1)$$

The indices  $\alpha$  and  $\beta$  refer to atomic nuclei and the indices *i* and *j* refer to electrons. The electron has charge -e and mass  $m_e$ ; the nuclei have charge  $Z_{\alpha}e$  and mass  $M_{\alpha}$ . The first summation is the kinetic energy of all the atomic nuclei,  $\hat{T}_N$ ; the second term is the total kinetic energy of the electrons,  $\hat{T}_e$ . The remaining terms are the potential-energy terms for nucleus-nucleus repulsion,  $V_{NN}$ , nucleus-electron attraction,  $V_{eN}$ , and electron-electron repulsion,  $V_{ee}$ . We have introduced a shorthand notation for the separation distance between two particles. E.g.,

$$R_{\alpha\beta} = \left| \mathbf{R}_{\alpha} - \mathbf{R}_{\beta} \right| = \sqrt{\left( X_{\alpha} - X_{\beta} \right)^{2} + \left( Y_{\alpha} - Y_{\beta} \right)^{2} + \left( Z_{\alpha} - Z_{\beta} \right)^{2}} , \qquad (1.2)$$

where  $\mathbf{R}_{\alpha} = (X_{\alpha}, Y_{\alpha}, Z_{\alpha})$  is the position of the  $\alpha^{\text{th}}$  nucleus. Henceforth, we denote the sets of nuclear and electronic coordinates as  $\{\mathbf{R}_{\alpha}\}$  and  $\{\mathbf{r}_{i}\}$ , respectively. We often decompose the molecular Hamiltonian into its nuclear and electronic terms,

$$\hat{H} = \hat{T}_{N}\left(\left\{\mathbf{R}_{\alpha}\right\}\right) + \hat{V}_{NN}\left(\left\{\mathbf{R}_{\alpha}\right\}\right) + \hat{H}_{ele}\left(\left\{\mathbf{r}_{i}\right\}, \left\{\mathbf{R}_{\alpha}\right\}\right), \qquad (1.3)$$

where the electronic Hamiltonian is defined as

$$\hat{H}_{ele}\left(\left\{\mathbf{r}_{i}\right\},\left\{\mathbf{R}_{\alpha}\right\}\right) = \hat{T}_{e}\left(\left\{\mathbf{r}_{i}\right\}\right) + \hat{V}_{eN}\left(\left\{\mathbf{r}_{i}\right\},\left\{\mathbf{R}_{\alpha}\right\}\right) + \hat{V}_{ee}\left(\left\{\mathbf{r}_{i}\right\}\right)$$
(1.4)

All the properties of a molecule can be determined from the molecular wavefunction, which is determined by solving the molecular Schrödinger equation,

$$\hat{H}\Psi(\{\mathbf{r}_i\},\{\mathbf{R}_{\alpha}\}) = E_{\text{total}}\Psi(\{\mathbf{r}_i\},\{\mathbf{R}_{\alpha}\})$$
(1.5)

In practice, it is only possible to accurately solve the molecular Schrödinger equation for very small molecules. Addressing the molecules of greatest interest to experimental chemists requires approximations.

Because nuclei are much heavier than electrons, the electrons in a molecule move much faster than its nuclei. If we assume that the electrons instantly adjust to any change in nuclear position (that is, that the electrons remain in a stationary state when the nuclear positions change), then we can neglect the dynamical correlation between the electrons and nuclei. That is, from the viewpoint of the electrons, the slow-moving nuclei are fixed. We can therefore neglect the nuclear kinetic energy when solving for the electronic portion of the wavefunction,

$$\left(\hat{H}_{ele}\left(\{\mathbf{r}_{i}\},\{\mathbf{R}_{\alpha}\}\right)+V_{NN}\left(\{\mathbf{R}_{\alpha}\}\right)\right)\psi_{ele}\left(\{\mathbf{r}_{i}\},\{\mathbf{R}_{\alpha}\}\right)=U\left(\{\mathbf{R}_{\alpha}\}\right)\psi_{ele}\left(\{\mathbf{r}_{i}\},\{\mathbf{R}_{\alpha}\}\right)$$
(1.6)

The atomic nuclei then move on the molecular potential energy surface,  $U(\{\mathbf{R}_{\alpha}\})$ , due to the cloud of electrons that surround them,

$$\left(\hat{T}_{N}\left(\left\{\mathbf{R}_{\alpha}\right\}\right)+U\left(\left\{\mathbf{R}_{\alpha}\right\}\right)\right)\chi_{N}\left(\left\{\mathbf{R}_{\alpha}\right\}\right)=E_{\text{total}}\chi_{N}\left(\left\{\mathbf{R}_{\alpha}\right\}\right)$$
(1.7)

and the molecular wavefunction is approximately

$$\Psi(\{\mathbf{r}_i\},\{\mathbf{R}_{\alpha}\}) \approx \chi_N(\{\mathbf{R}_{\alpha}\}) \Psi_{ele}(\{\mathbf{r}_i\},\{\mathbf{R}_{\alpha}\})$$
(1.8)

Equations (1.6)-(1.8) define the Born-Oppenheimer approximation.<sup>12</sup>

Because the nucleus-nucleus repulsion energy does not depend on the electronic positions, it can be removed from Eq. (1.6). The remaining terms define the electronic Schrödinger equation,

$$\hat{H}_{ele}(\{\mathbf{r}_i\},\{\mathbf{R}_{\alpha}\})\psi_{ele}(\{\mathbf{r}_i\},\{\mathbf{R}_{\alpha}\}) = E_{ele}(\{\mathbf{R}_{\alpha}\})\psi_{ele}(\{\mathbf{r}_i\},\{\mathbf{R}_{\alpha}\}).$$
(1.9)

Adding the nucleus-nucleus repulsion to the electronic energy recovers the potential energy surface,

$$U(\{\mathbf{R}_{\alpha}\}) = E_{ele}(\{\mathbf{R}_{\alpha}\}) + V_{NN}(\{\mathbf{R}_{\alpha}\}).$$
(1.10)

Computationally, the molecular potential energy surface is determined by repeatedly solving the electronic Schrödinger equation, Eq. (1.9), for specific choices of the nuclear positions. Solving the electronic Schrödinger equation is the task of quantum chemistry and, more specifically, molecular electronic structure theory. We will not review the pantheon of quantum chemistry methods and basis sets that are available, but merely note that there are two main types of methods.<sup>10,11,13-15</sup> Mean-field models like Hartree-Fock and Kohn-Sham density functional theory (DFT) are based on the orbital picture, and each electron is considered to occupy an orbital, and more quasi-independently in the effective potential defined by the atomic nuclei and the other electrons in the molecule. In post-mean-field (often called post-Hartree-Fock) approaches like the configuration interaction, coupled cluster, and Møller-Plesset perturbation theory

methods, the electron-electron repulsion is accounted for in a more detailed way, and the wavefunction includes contributions from many, partially occupied, orbitals.

For a molecule with *N* nuclei, the potential energy surface is a real-valued function whose domain is the 3*N*-dimensional space of nuclear positions. However, because the energy of a molecule doesn't depend on its position or orientation in space, only 3N - 6 (3N - 5 for a linear molecule) coordinates suffice to determine the molecular conformation and therefore the value of the molecular potential energy surface. That is, the value of the molecular potential energy is invariant to translation (3 dimensions) and rotation (3 dimensions in general, but only 2 for a linear molecule) of the molecule. The molecular potential energy surface is therefore viewed as a function of the 3N - 6 (or 3N - 5) internal degrees of freedom of the molecule.

### **B.** Characterization of Potential Energy Surfaces

A molecule's potential energy surface (PES) represents how the energy of the molecule depends on its geometric conformation. The PES defines an energy landscape with valleys (stable molecular conformations) connected by mountain passes (transition states). Stationary points on the PES, where  $\nabla U(\{\mathbf{R}_{\alpha}\}) = \mathbf{0}$ , are the key structures that one typically uses to characterize and understand the chemical reaction mechanism.<sup>16</sup>

Structures at the bottom of valleys (local minima in  $U(\{\mathbf{R}_{\alpha}\})$ ) are chemically stable structures: reactants, products, and stable intermediates. With rare exceptions, the minimum-energy path between the reactant and product of an elementary reaction goes through a first-order saddle point; the molecular conformation at this saddle point is labeled as the transition state structure for the reaction. (At an  $n^{\text{th}}$ -order saddle point the PES is a maximum in *n* directions and a minimum in the other 3N - 6 - n (or 3N - 5 - n) directions. That is, the second derivative matrix of the PES,  $\nabla \nabla^T U(\{\mathbf{R}_{\alpha}\})$ , has *n* negative eigenvalues and the remaining eigenvalues are positive).

Denoting the set of all the nuclear coordinates that are used to specify the molecular conformation as  $\mathbf{x}$ , the gradient of the energy is a vector,

$$\mathbf{g}(\mathbf{x}) = \nabla U(\mathbf{x}) = \begin{bmatrix} \frac{\partial U(\mathbf{x})}{\partial x_1} \\ \frac{\partial U(\mathbf{x})}{\partial x_2} \\ \vdots \end{bmatrix}.$$
(1.11)

At the point **x**, the atomic nuclei feel a force  $\mathbf{F}(\mathbf{x}) = -\mathbf{g}(\mathbf{x})$  pushing them towards the bottom of the nearest potential-energy well. At a stationary point on the PES,  $\mathbf{g}(\mathbf{x}) = \mathbf{0}$  and no atomic nucleus feels a force in any direction.

The second derivative matrix, called the Hessian of the PES, is the matrix of force constants,

$$\mathbf{H}(\mathbf{x}) = \nabla \nabla^{T} U(\mathbf{x}) = \begin{bmatrix} \frac{\partial^{2} U}{\partial x_{1}^{2}} & \frac{\partial^{2} U}{\partial x_{1} \partial x_{2}} & \cdots \\ \frac{\partial^{2} U}{\partial x_{2} \partial x_{1}} & \frac{\partial^{2} U}{\partial x_{2}^{2}} & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix}.$$
(1.12)

The generalized eigenvectors of the Hessian, in the metric defined by the atomic mass matrix, are the normal modes of vibrational motion.<sup>17</sup> At a local minimum on the PES, all

the eigenvalues of the Hessian are positive and all the vibrational frequencies are real. At a transition state, exactly one eigenvalue of the Hessian is negative and there is one imaginary vibrational frequency corresponding to a vector that is tangent to the reaction path at the transition state. Chemically, this vector indicates the reaction coordinate for the barrier-crossing reactive event.

The PES provides fundamental insight into chemical reactions. Figure 1.1 shows an example of 4-well 2-dimensional potential energy surface.<sup>18</sup> In this figure, there are four different (meta)stable structures (the minima), and four transition states (the saddle-points) connecting them. The fundamental thermodynamic and kinetic properties of a reaction can be computed from its potential energy surface. The difference in energy between two minima (reactant and product) is the reaction energy;<sup>3</sup> this reaction (free) energy determines the equilibrium constant. The energy difference between a minimum and a first-order saddle point (reactant and transition state) defines the reaction's energy barrier. Using transition state theory, the rate of an elementary reaction can be approximated from the reaction barrier height. Finally, the plausible reaction mechanisms can be determined by finding all the low-energy reaction pathways from the reactant structure, through transition state(s) and possible reactive intermediates, to the product structure.


**Figure 1.1** A four-well two-dimensional potential energy surface,<sup>18</sup> showing four minima and four saddle points.

# C. Numerical Calculations Using Potential Energy Surfaces

For geometry optimization, most methods use only the value of the energy and its first derivative, though some methods use the second derivative as well. In general, optimization methods that depend on the derivatives of the potential energy function require fewer iterations than those depending only on the function value. Moreover, the computational cost per iteration tends to be similar because, for most approximate methods, the cost of computing the gradient is comparable to the cost of computing the energy.<sup>9</sup> Therefore, most optimization methods rely on the gradient. Mathematical

formulas for evaluating the Hessian (second-derivative) are available for most of levels of theory, but the cost-of-computation is much higher than it is for the energy and the gradient.

In real chemical reactions, reactant molecules do not transform to product by a single pathway. Instead, there is an ensemble of trajectories<sup>19</sup>. Calculating a representative sample of these trajectories is computationally expensive, so a single representative path is commonly used. The minimum energy path, MEP, describes the lowest-energy way for reactant molecule, interacting with its environment, to become product. In an elementary (single-step) reaction, the MEP is just the steepest-descent path from the transition state to the reactant and the product. Because usually only the end points of the MEP are known initially, determining the MEP without first computing the transition state is mathematically and computationally difficult (but not impossible).

The steepest descent path,  $\mathbf{x}(s)$ , is defined as the solution to the system of differential equations,

$$\frac{d\mathbf{x}(s)}{ds} = -\frac{\mathbf{g}(\mathbf{x}(s))}{|\mathbf{g}(\mathbf{x}(s))|} = -\hat{\mathbf{g}}(\mathbf{x}(s)), \qquad (1.13)$$

where *s* is the arc length of the path. Standard methods for the numerical integration of differential equations can be used to solve Eq. (1.13), but the equations are often ill-conditioned, especially when the path is highly curved. In such cases, conventional approaches sometimes converge to spurious "kinked" solutions to the path.<sup>20,21</sup> Several methods for finding the steepest descent path are in common use. Examples are the (a) Ishida, Morokuma, Komornicki (IMK) method,<sup>22</sup> which uses an

Euler step with line search, (b) the LQA method<sup>23,24</sup> of Mclver and Page, which solves equation (1.13) analytically, and (c) the second-order implicit trapezoid method of Gonzalez and Schlegel (GS2).<sup>25,26</sup> Burger and Yang generalized the implicit trapezoid method to create a family of implicit-explicit methods.<sup>27,28</sup>

# **1.3** Coordinate Systems for Expressing the Potential Energy Surface (PES)

### A. Cartesian Coordinate System

The choice of coordinate system has a large effect on the efficiency of geometry optimization. In general any set of coordinates that provides a complete description of the system can be used. However, the structure of the Hessian matrix of the potential energy, Eq. (1.12), depends strongly on the choice of coordinate system, and this greatly influences the efficiency of optimization, especially when the Hessian matrix must be approximated.<sup>29</sup>

Conceptually, it is simplest to use the Cartesian coordinates, comprising the  $\{(X_{\alpha}, Y_{\alpha}, Z_{\alpha})\}$  coordinates of each atomic nucleus,  $\alpha$ . For a molecule with *N* atoms, there are 3*N* Cartesian coordinates. Cartesian coordinates are straightforward to use because methods for approximating the electronic energy and its derivatives usually use Cartesian coordinates. Their disadvantage is that the Cartesian coordinates are highly coupled: chemical changes in molecular structure (e.g., stretching a bond) typically induce changes

in almost all of the molecule's Cartesian coordinates. The Hessian matrix has large offdiagonal elements reflecting these couplings, which makes it difficult to approximate. This makes it difficult to build a good quadratic approximation to the potential energy surface.

# **B.** Systems of Redundant and Nonredundant Internal Coordinates

Internal coordinates, comprising bond lengths, bond angles, and dihedral angles between bonds, are chemically intuitive and give mathematically compact descriptions of chemically important molecular motions. These coordinates are less coupled, so there are fewer off-diagonal elements in the Hessian, making it easier to approximate.

Since translation and rotation of molecules does not affect the internal coordinates, this system of coordinates might be assumed to give potential energy surfaces of the desired 3N - 6 (3N - 5 for a linear molecule) dimensionality. This is not the case, however, because for molecules with more than two atoms, the number of internal coordinates far exceeds the number of internal degrees of freedom that a molecule has.<sup>4,6-8,30-41</sup> This is called the redundancy problem. For example, in a nonlinear triatomic molecule, there are three internuclear distances ( $R_{\alpha\beta}$ ,  $R_{\alpha\gamma}$ ,  $R_{\beta\gamma}$ ) and three unique bond angles ( $\angle \alpha\beta\gamma$ ,  $\angle \beta\alpha\gamma$ , and  $\angle \beta\gamma\alpha$ ). Choosing one of the bond lengths, and any two of the remaining five coordinates, fully specifies the molecular geometry. For a nonlinear triatomic molecule, there are 20 equivalent ways to express the molecular geometry in internal coordinates. For molecules with more atoms, the redundancy problem is even more severe.

The redundancy problem can be solved by arbitrarily omitting specific coordinates to form a non-redundant set of internal coordinates. This is done in the Z-matrix approach,<sup>42</sup> which works reasonably well in most cases. However, leaving out specific coordinates can give strong linear and nonlinear couplings between coordinates, and determining the best coordinates to include is difficult. In a triatomic molecule, the Zmatrix includes two bond lengths (e.g.,  $R_{\alpha\beta}$  and  $R_{\beta\gamma}$ ) and the angle between them ( $\angle \alpha\beta\gamma$ ). This means that changes in the bond length of the unspecified side of the three-atom ring has a complicated mathematical description

$$(R_{\alpha\gamma} = \sqrt{(R_{\alpha\beta}\cos(\angle\alpha\beta\gamma) - R_{\beta\gamma})^2 + R_{\alpha\beta}^2\sin^2(\angle\alpha\beta\gamma))}$$
 that couples all of the other coordinates describing the ring. Unsurprisingly, in some cases, and especially for molecules containing rings, the Z-matrix is not the best choice, and even Cartesian coordinates often perform better.<sup>43,44</sup>

The natural internal coordinates are a set of nonredundant internal coordinates based on local pseudosymmetry.<sup>30,45</sup> In this set, bonds are individual stretching coordinates, but linear combinations of angles and dihedrals are chosen to describe bends and torsions (dihedrals) of a system. Unfortunately, generating natural internal coordinates automatically is complicated, and particular difficult for molecules containing rings—precisely the case where they are most helpful.

To remedy this problem, Pulay formulated an approach for geometry optimization using the redundant coordinates directly.<sup>6</sup> (In his early work, he used all possible bonds, angles, and dihedrals; this is called the system of primitive redundant internal coordinates. Nowadays, subsets of this primitive redundant internal coordinates are often used.) Baker showed how redundant internal coordinates can be simply reduced to a nonredundant set of coordinates called delocalized internal coordinates.<sup>31,33</sup> The delocalized internal coordinates are linear combinations of the primitive redundant set; they represent a computationally practical alternative to Pulay's natural internal coordinates.

## C. Transforming Between Internal and Cartesian Coordinate Systems

Because the electronic energy and its derivatives are usually computed using Cartesian coordinates, using (redundant) internal coordinates in geometry optimization requires methods for converting Cartesian coordinates to internal coordinates, and vice versa. The internal coordinates,  $\{q_i\}$ , are nonlinear functions of the Cartesian coordinates,  $\{x_j\}$ , and the Jacobian of the transformation is the Wilson **B** matrix, with elements<sup>17</sup>

$$b_{ij} = \frac{\partial q_i}{\partial x_j} \tag{1.14}$$

The Wilson **B** matrix is rectangular; it has one row for each of the  $N_{int}$  internal coordinate and one column for each of the 3N Cartesian coordinates. Note that the transformation from Cartesian coordinates to redundant internal coordinates is not invertible. For example, there is no molecule (ergo no set of Cartesian coordinates) for which  $\angle \alpha\beta\gamma + \angle \beta\alpha\gamma + \angle \beta\gamma\alpha \neq 180$  degrees.

Using the Wilson matrix, an infinitesimal change in Cartesian coordinates can be converted to the corresponding change in internal coordinates,<sup>4</sup>

$$\delta \mathbf{q} = \mathbf{B} \cdot \delta \mathbf{x} \tag{1.15}$$

Since it is easy to compute the value of the internal coordinates for any given set of Cartesian coordinates, Eq. (1.15) is rarely used except as an intermediate step in mathematical derivations. The change in Cartesian coordinates from an infinitesimal change in internal coordinates is computed as

$$\mathbf{B}^+ \boldsymbol{\delta} \mathbf{q} = \boldsymbol{\delta} \mathbf{x} \tag{1.16}$$

 $\mathbf{B}^{+}$  is the Moore-Penrose pseudo-inverse of the rectangular matrix  $\mathbf{B}$ .

Using the **B** matrix and its inverse, the gradient and the Hessian can be converted to Cartesian coordinates from internal coordinates,

$$\mathbf{g}_{x} = \mathbf{B}^{T} \mathbf{g}_{a} \tag{1.17}$$

$$\mathbf{H}_{x} = \mathbf{B}^{T} \mathbf{H}_{a} \mathbf{B} + \mathbf{K}$$
(1.18)

and from Cartesian coordinates to internal coordinates,

$$\mathbf{g}_q = \left(\mathbf{B}^T\right)^+ \mathbf{g}_x \tag{1.19}$$

$$\mathbf{H}_{q} = \left(\mathbf{B}^{T}\right)^{+} \left(\mathbf{H}_{x} - \mathbf{K}\right)\mathbf{B}^{+}$$
(1.20)

The elements of the matrix  $\mathbf{K}$  involve the second derivative of the internal coordinates with respect to the Cartesian coordinates,

$$k_{jk} = \sum_{i=1}^{N_{int}} \left[ \mathbf{g}_{q} \right]_{i} \frac{\partial^{2} q_{i}}{\partial x_{j} \partial x_{k}} = \sum_{i=1}^{N_{int}} \left[ \mathbf{g}_{q} \right]_{i} \frac{\partial b_{ij}}{\partial x_{k}}$$
(1.21)

For small (but not infinitesimal) change in internal coordinates, Eq. (1.16) is unreliable. Since it is not simple, or even always possible, to convert a change in internal coordinates to a change in Cartesian coordinates, iterative procedures are typically used.<sup>4,6-8,30-41</sup>

Suppose we wish to find the set of Cartesian coordinates,  $\mathbf{x}_{target}$  for a given set of internal coordinates,  $\mathbf{q}_{target}$ . Let  $\mathbf{x}_0$  denote an initial set of Cartesian coordinates and  $\mathbf{q}_0$  denote the corresponding set of internal coordinates. We can estimate the target Cartesian coordinates using Eq. (1.16),

$$\mathbf{x}_1 = \mathbf{x}_0 + \mathbf{B}^+ \mathbf{s}_a \tag{1.22}$$

where  $\mathbf{s}_q$  is the step in internal coordinates,

$$\mathbf{s}_{q} = \mathbf{q}_{\text{target}} - \mathbf{q}_{0} \tag{1.23}$$

To assess the accuracy of Eq. (1.22), the set of internal coordinates corresponding to the new Cartesian coordinates is computed,  $\mathbf{q}_1 = \mathbf{q}(\mathbf{x}_1)$  and the discrepancy between this value and the target internal coordinates is assessed,

$$\Delta \mathbf{q} = \mathbf{q}_{\text{target}} - \mathbf{q}_{1}. \tag{1.24}$$

If this discrepancy is sufficiently small, then  $\mathbf{x}_{target} = \mathbf{x}_1$ . Otherwise, one repeats the procedure. I.e., for k = 1, 2, ..., one computes a new estimate for the Cartesian coordinates,

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{B}^+ \Delta \mathbf{q}_k \tag{1.25}$$

where

$$\Delta \mathbf{q}_{k} = \mathbf{q}_{\text{target}} - \mathbf{q}_{k}$$
  
$$\mathbf{q}_{k} = \mathbf{q}(\mathbf{x}_{k})$$
 (1.26)

until  $\Delta \mathbf{q}_k$  is sufficiently small. Unfortunately, this procedure does not always converge. To ensure that the iterations eventually terminate, whenever  $|\Delta \mathbf{q}_k| > |\Delta \mathbf{q}_1|$ , the initial estimate obtained by naïvely using Eq. (1.16) is used instead. This most commonly occurs when the target internal coordinates are very different from the initial internal coordinates, when redundant internal coordinates are used, or when there are nearly linear bond angles.

We have proposed a different approach for transforming internal to Cartesian coordinates. We simply choose the set of Cartesian coordinates whose associated internal coordinates are as close as possible to the target internal coordinates,

$$\min_{\mathbf{x}} \left( \mathbf{q}_{\text{target}} - \mathbf{q}(\mathbf{x}) \right)^T \mathbf{W} \left( \mathbf{q}_{\text{target}} - \mathbf{q}(\mathbf{x}) \right)$$
(1.27)

W is a positive-definite diagonal matrix; if W were the identity matrix, this would merely minimize the (squared) distance between the target internal coordinates and the internal coordinates computed from the Cartesian coordinates. However, we found that it was better to weight different types of internal coordinates (bond lengths, bond angles, dihedral angles) differently. We use  $\mathbf{x} = \mathbf{x}_0 + \mathbf{B}^+ (\mathbf{q}_{target} - \mathbf{q}_0)$  as the initial guess for this minimization (just as in the iterative approach).

Geometrically,  $\mathbf{q}(\mathbf{x})$  is a 3N - 6 (or 3N - 5) dimensional manifold embedded in a space with the dimensionality of the internal coordinates,  $N_{\text{dim}}$ . An arbitrary point in the space of internal coordinates,  $\mathbf{q}_{\text{target}}$ , may or may not lie on the manifold of physical molecular geometries,  $\mathbf{q}(\mathbf{x})$ . If  $\mathbf{q}_{\text{target}}$  is on the manifold, Eq. (1.27) finds a corresponding

set of Cartesian coordinates (which is unique except for translation and rotation). If  $\mathbf{q}_{\text{target}}$  is not on the manifold. Eq. (1.27) finds the closest point on the manifold, and uses the Cartesian coordinates from that location as the best possible representation for  $\mathbf{q}_{\text{target}}$ .

### **D.** The Problem of Linear Dihedrals

In the previous section, we assumed that a given set of Cartesian coordinates gives rise to a unique set of internal coordinates. When this is not true, the mapping  $\mathbf{q}(\mathbf{x})$  is not well-defined and the **B** matrix has an infinite eigenvalue, because an infinitesimal change in Cartesian coordinates can cause an arbitrarily large change in internal coordinates.

The mapping from Cartesian to internal coordinates is only ill-defined when one of the internal coordinates is a dihedral angle (torsion)  $\angle \alpha\beta\gamma\delta$  and one of the bond angles  $\angle \alpha\beta\gamma$  or  $\angle \beta\gamma\delta$  is 180° (the three atoms are collinear). To define the dihedral angle mathematically, one first defines the vectors aligned with the bonds between the atoms,

$$\begin{aligned} \mathbf{b}_{\alpha\beta} &= \mathbf{R}_{\alpha} - \mathbf{R}_{\beta} \\ \mathbf{b}_{\beta\gamma} &= \mathbf{R}_{\beta} - \mathbf{R}_{\gamma} \\ \mathbf{b}_{\gamma\delta} &= \mathbf{R}_{\delta} - \mathbf{R}_{\gamma} \end{aligned} \tag{1.28}$$

The dihedral angle is the angle between the plane formed by the first bond and the central bond ( $\mathbf{b}_{\alpha\beta}$  and  $\mathbf{b}_{\beta\gamma}$ ) and the plane formed by the last bond and the central bond ( $\mathbf{b}_{\gamma\delta}$  and  $\mathbf{b}_{\beta\gamma}$ ). If either the first two vectors or the last two vectors are collinear (i.e., if  $\angle \alpha\beta\gamma$  or  $\angle \beta\gamma\delta$  is zero or 180°), then they fail to define a plane and the dihedral angle is ill-defined.

In practice, when either the first two or the last two bond vectors are nearly collinear, a large change in dihedral angle causes only a very small change in the Cartesian coordinates. I.e., the mapping  $\mathbf{q}(\mathbf{x})$  is numerically ill-conditioned; this can cause algorithms based on internal coordinates to fail. In addition, sometimes all the dihedral angles are well-defined in the initial structure, but a linear (or near-linear) dihedral arises during the course of the optimization process, causing the dihedral angle to become ill-defined and the algorithm to fail.

To circumvent these problems, we defined two new coordinates,

$$\frac{\mathbf{b}_{\alpha\beta} \cdot \mathbf{b}_{\gamma\delta}}{\left|\mathbf{b}_{\alpha\beta}\right| \left|\mathbf{b}_{\gamma\delta}\right|} = \hat{\mathbf{b}}_{\alpha\beta} \cdot \hat{\mathbf{b}}_{\gamma\delta}$$
(1.29)

$$\frac{\mathbf{b}_{\beta\gamma} \cdot \left(\mathbf{b}_{\alpha\beta} \times \mathbf{b}_{\gamma\delta}\right)}{\left\|\mathbf{b}_{\alpha\beta}\right\| \left\|\mathbf{b}_{\beta\gamma}\right\| \left\|\mathbf{b}_{\gamma\delta}\right\|} = \hat{\mathbf{b}}_{\beta\gamma} \cdot \left(\hat{\mathbf{b}}_{\alpha\beta} \times \hat{\mathbf{b}}_{\gamma\delta}\right)$$
(1.30)

which we use in lieu of conventional dihedral angles. The first function is the 1-4 cosine, while the second function is the volume of the parallelepiped defined by the bond vectors. Unlike the dihedral angle, these coordinates are well-defined even when  $\mathbf{b}_{\beta\gamma}$  is collinear with one of the other vectors.

# **1.4** Numerical Methods for Optimization

#### A. Newton's Method

Given an initial molecular structure with the coordinates  $\mathbf{x}_0$ , it is usually only practical to compute the potential energy of the structure,  $U(\mathbf{x}_0)$ , as well as some loworder derivatives of the PES at  $\mathbf{x}_0$ . The goal of a local geometry optimization method is to exploit this available information to find a new structure,  $\mathbf{x}_1$ , which is closer to the nearest local minimum on the PES (for geometry minimization) or the nearest saddle point on the PES (for transition-state optimization). To do this, one develops the potential energy surface as a Taylor series,

$$U(\mathbf{x}) = U(\mathbf{x}_0) + \mathbf{g}_0 \cdot (\mathbf{x} - \mathbf{x}_0) + \frac{1}{2} (\mathbf{x} - \mathbf{x}_0)^T \mathbf{H}_0 (\mathbf{x} - \mathbf{x}_0) + \cdots$$
(1.31)

Because higher-order derivatives are increasingly demanding to compute, this expansion is usually (but not always<sup>46</sup>) truncated after the first- or second-order terms. At a stationary point on a potential energy surface, the gradient is zero. Using the Taylor series for the gradient gives the equation,

$$\mathbf{0} = \mathbf{g}(\mathbf{x}) = \mathbf{g}_0 + \mathbf{H}_0(\mathbf{x} - \mathbf{x}_0) + \cdots$$
(1.32)

Defining the step,  $\Delta \mathbf{x} = \mathbf{x} - \mathbf{x}_0$ , and truncating Eq. (1.32) after the second-derivative term, one has the system of linear equations,

$$\mathbf{H}_{0}\Delta\mathbf{x} = -\mathbf{g}_{0} \tag{1.33}$$

with formal solution

$$\Delta \mathbf{x} = -\mathbf{H}_0^{-1} \mathbf{g}_0 \tag{1.34}$$

This is called the Newton step. If one uses the Newton step to update the structure,  $\mathbf{x}_i = \mathbf{x}_i$ .  $I + \Delta \mathbf{x}$ , and computes the gradient,  $\mathbf{g}(\mathbf{x}_i) = \mathbf{g}(\mathbf{x}_{i-1} + \Delta \mathbf{x})$ , the gradient will not be exactly zero because the PES is not exactly quadratic. One must then iterate the procedure by repeatedly solving the system of linear equations,

$$\mathbf{H}_{i}\Delta\mathbf{x} = -\mathbf{g}_{i},\tag{1.35}$$

until the gradient is zero. This is Newton's method of optimization. Almost all numerical optimization methods are embellishments of Newton's method.<sup>47-51</sup>

Conceptually, in Newton's method one fits the PES at the location of the current structure,  $\mathbf{x}_i$ , with a paraboloid, and then solves for the stationary point (either a minimum, a maximum, or a saddle point) of the paraboloid. When the  $\Delta \mathbf{x}$  predicted by Newton's method is too big, higher-derivative terms in Eqs. (1.31) are nonnegligible, the paraboloid is no longer a good approximation to the PES, and Newton's method is unreliable. In order to ensure that Newton's method converges, one must not allow steps that stray from the region of the PES where the quadratic approximation in Eq. (1.31) is trustworthy. This is often achieved by imposing a trust radius: one attempts to solve Eq. (1.35) but if  $|\Delta \mathbf{x}|$  is too large, one chooses  $\Delta \mathbf{x}$  so that the gradient

$$\mathbf{g}(\mathbf{x}_i + \Delta \mathbf{x}) = \mathbf{g}_i + \mathbf{H}_i \Delta \mathbf{x}$$
(1.36)

is as small as possible, subject to the constraint that  $|\Delta \mathbf{x}| < \tau$ . The trust radius,  $\tau$ , is chosen so that the quadratic approximation is valid and it is updated throughout the calculation. As an alternative to the trust-radius approach, one can consider a more general update,  $\mathbf{x}_i$  $= \mathbf{x}_{i-1} + \alpha \Delta \mathbf{x}$ , where  $\alpha$  is optimized to ensure that the magnitude of the gradient decreases enough to guarantee convergence. This is called the line-search method. Both trust-radius and line-search methods suffice to ensure the convergence of Newton's method; they are discussed in more detail in section 1.4.C.

## **B.** Quasi-Newton Methods

Newton's method requires evaluating the second-derivative matrix, or Hessian, of the PES at each iteration. Computing the Hessian is computationally prohibitive for large molecules and the most accurate post-Hartree-Fock quantum chemistry methods, and computationally expensive even for moderately-sized molecules with standard quantum-chemistry methods. This motivates interest in quasi-Newton methods, where the Hessian is approximated using the gradient-calculations from successive steps of the optimization.<sup>47-51</sup> Because the gradient of the PES can typically be computed with little computational cost beyond computing the value of the PES itself, quasi-Newton methods are much more efficient.

In quasi-Newton methods, one starts with an initial Hessian, which can be computed either exactly or approximately. If the initial Hessian is approximated, one often uses semiempirical methods<sup>52,53</sup> or an inexpensive calculation, like Hartree-Fock with a minimal basis set. (In Chapter 5, we show that the latter approach is quite effective.) Sometimes one uses a Hessian based on a molecular mechanics force field, introducing terms for the stretching-motions, angle-bending, and torsional modes of a molecule.<sup>54-56</sup> In the absence of available information, one can use a scaled identity matrix for the Hessian, but starting with this initial Hessian usually leads to a relatively slow optimization procedure.<sup>57</sup>

At each step of the procedure, the Hessian is updated using the values and gradients of the PES evaluated in previous steps. There are dozens of quasi-Newton updates for the Hessian, but the defining feature of any quasi-Newton method is the secant condition, which ensures that the gradient of the quadratic model constructed with the updated Hessian matches the computed gradient. I.e.,

$$\mathbf{g}^{\text{new}} - \mathbf{g}^{\text{old}} = \mathbf{H}^{\text{new}} \left( \mathbf{x}^{\text{new}} - \mathbf{x}^{\text{old}} \right)$$
(1.37)

(Compare Eq. (1.32).) The change in gradient between two iterations is conventionally denoted

$$\mathbf{y} = \mathbf{g}^{\text{new}} - \mathbf{g}^{\text{old}} \tag{1.38}$$

The Broyden-Fletcher-Goldfarb-Shanno (BFGS) method is one of most popular and effective quasi-Newton methods<sup>58-62</sup>

$$\mathbf{H}^{\text{new}} = \mathbf{H}^{\text{old}} + \frac{\mathbf{y}\mathbf{y}^{T}}{\mathbf{y}\cdot\Delta\mathbf{x}} - \frac{\left(\mathbf{H}^{\text{old}}\Delta\mathbf{x}\right)\left(\mathbf{H}^{\text{old}}\Delta\mathbf{x}\right)^{T}}{\left(\Delta\mathbf{x}\right)^{T}\mathbf{H}^{\text{old}}\Delta\mathbf{x}}$$
(1.39)

The BFGS method ensures that the Hessian is positive definite (provided the initial Hessian is also positive-definite), so BFGS is an effective quasi-Newton approach for geometry minimization, but not for transition-state optimization.

The BFGS is the most famous member of a group of quasi-Newton methods that often referred to as the Broyden family. All members of the Broyden family are rank-two updates, since the old Hessian is updated with two rank-one corrections in Eq. (1.39). All of the quasi-Newton updates in the Broyden family are positive definite. There are simpler rank-one updates, the most popular of which is the symmetric-rank-1 (SR1) update,<sup>57</sup>

$$\mathbf{H}^{\text{new}} = \mathbf{H}^{\text{old}} + \frac{\left(\mathbf{y} - \mathbf{H}^{\text{old}}\Delta\mathbf{x}\right)\left(\mathbf{y} - \mathbf{H}^{\text{old}}\Delta\mathbf{x}\right)^{T}}{\left(\mathbf{y} - \mathbf{H}^{\text{old}}\Delta\mathbf{x}\right) \cdot \Delta\mathbf{x}}$$
(1.40)

The SR1 update is known for providing better approximations to the true Hessian than more-sophisticated rank-two updates like BFGS. Unlike the BFGS Hessian, the SR1 update does not preserve positive-definiteness of the Hessian. This makes SR1 less suitable for geometry minimization, but preferable for transition-state optimization. However, because the SR1 update can be numerically unstable when  $|\mathbf{y} - \mathbf{H}\Delta\mathbf{x}|$  is small, it is seldom used.

The Powell-symmetric-Broyden (PSB) update combines the accuracy of the ranktwo updates (like BFGS) with the non-positive-definiteness of the SR1 update (useful for transition-state optimization). It is,<sup>49</sup>

$$\mathbf{H}^{\text{new}} = \mathbf{H}^{\text{old}} + \frac{\left(\mathbf{y} - \mathbf{H}^{\text{old}}\Delta\mathbf{x}\right)\Delta\mathbf{x}^{T} + \Delta\mathbf{x}\left(\mathbf{y} - \mathbf{H}^{\text{old}}\Delta\mathbf{x}\right)^{T}}{\left|\Delta\mathbf{x}\right|^{2}} - \left(\frac{\Delta\mathbf{x}\cdot\left(\mathbf{y} - \mathbf{H}^{\text{old}}\Delta\mathbf{x}\right)}{\left|\Delta\mathbf{x}\right|^{4}}\right)\Delta\mathbf{x}\Delta\mathbf{x}^{T}.$$
(1.41)

The SR1 and PSB methods are both appropriate for transition-state optimization. However, as Bofill showed, the linear combination of the two updates is even better for transition-state optimization,<sup>63</sup>

$$\mathbf{H}^{\text{new}} = \left(1 - \phi\right) \mathbf{H}^{\text{new}}_{\text{SR1}} + \phi \mathbf{H}^{\text{new}}_{\text{PSB}}$$
(1.42)

 $\varphi$  is the square of the sine of the angle between the step,  $\Delta \mathbf{x}$ , and the error in the old Hessian's approximation to the change in gradient that accompanies the step,

$$\phi = 1 - \frac{\left|\Delta \mathbf{x} \cdot \mathbf{E}\right|^2}{\left|\Delta \mathbf{x}\right|^2 \left|\mathbf{E}\right|^2}$$
(1.43)

$$\mathbf{E} = \mathbf{y} - \mathbf{H}^{\text{old}} \Delta \mathbf{x} \tag{1.44}$$

Later, Farkas and Schlegel proposed a similar update, using the SR1 and the BFGS (instead of the PSB) quasi-Newton updates.<sup>64</sup> Bofill then revisited the problem, proposing two new quasi-Newton methods that are suitable for transition state optimizations and/or minimizations.<sup>65</sup>

The effectiveness of quasi-Newton methods can be further improved by using one's chemical insight to identify the key chemical coordinates that are involved in a transition state. These are typically the coordinates associated with the bonds that are breaking or forming during the chemical reaction. As shown in reference <sup>66</sup> and Chapters 3 and 4, excellent results can be obtained if entries in the Hessian matrix corresponding to these "reduced coordinates" are accurately computed, e.g., using finite differences of gradient calculations. (Since the negative-curvature direction of the Hessian should be contained in the reduced subspace, the remaining block of the Hessian can be effectively approximated by the damped-BFGS quasi-Newton update.) Other methods for identifying key chemical coordinates and infusing the Hessian with extra information about them include the QST3 method<sup>67</sup> and a large family of "dimer methods."<sup>68,69</sup>

The mathematical derivation of the BFGS quasi-Newton update is invalid when the curvature condition,  $\mathbf{y} \cdot \Delta \mathbf{x} > 0$  is violated and the BFGS update is numerically illconditioned when  $\mathbf{y} \cdot \Delta \mathbf{x}$  is close to zero. The damped-BFGS method avoids this by replacing **y** in Eq. (1.39) with a new vector,<sup>70</sup>

$$\mathbf{r} = \mathbf{\Theta} \mathbf{y} + (1 - \mathbf{\Theta}) \mathbf{H}^{\text{old}} \Delta \mathbf{x}, \qquad (1.45)$$

where  $\Theta = \theta_i \delta_{ij}$  is a diagonal matrix containing the damping coefficients. When there are no reduced coordinates, the damping coefficients are<sup>57</sup>

$$\boldsymbol{\theta}_{i} = \begin{cases} 1 & \mathbf{y} \cdot \Delta \mathbf{x} \ge .2 (\Delta \mathbf{x})^{T} \mathbf{H}^{\text{old}} \Delta \mathbf{x} \\ \frac{.8 (\Delta \mathbf{x})^{T} \mathbf{H}^{\text{old}} \Delta \mathbf{x}}{(\Delta \mathbf{x})^{T} \mathbf{H}^{\text{old}} \Delta \mathbf{x} - \mathbf{y} \cdot \Delta \mathbf{x}} & \text{otherwise} \end{cases}$$

(1.46)

When there are reduced coordinates, the damped-BFGS procedure is updated so that only the non-reduced coordinates are affected.

In a transition-state optimization the Hessian should have one negative eigenvalue; for a geometry minimization, all the eigenvalues of the Hessian should be positive. Quasi-Newton updates sometimes give Hessians whose eigenvalues do not satisfy these requirements. In such cases, it is helpful to manually adjust the eigenvalue spectrum of the Hessian. Some methods for performing these eigenvalue shifts are presented in the next section.

# C. Step-Size Control

Given a computed value of the gradient and either the exact Hessian (Newton's method) or approximate Hessian (quasi-Newton methods), the direction of the next step in the optimization can be determined by solving the system of linear equations (1.35). The Hessian provides key information about the length that the step should be, but when the quadratic approximation to the PES is not reliable, (quasi-)Newton methods may not converge. To ensure convergence, one must ensure that the step does not stray outside the region where the quadratic approximation is reliable. That is, convergence requires a method to throttle back Newton steps that are too long.<sup>57</sup>

#### 1. Line search method:

In the line-search method, one uses the Newton step, (1.35), to determine the direction of the step, but one adjusts the length of the step to ensure that adequate progress towards a solution is made. Explicitly, one defines a unit vector in the direction of the Newton step,

$$\hat{\mathbf{p}} = -\frac{\left(\mathbf{H}^{\text{old}}\right)^{-1} \mathbf{g}^{\text{old}}}{\left|\left(\mathbf{H}^{\text{old}}\right)^{-1} \mathbf{g}^{\text{old}}\right|}$$
(1.47)

and considers steps of the form

$$\mathbf{x}^{\text{new}} = \mathbf{x}^{\text{old}} + \alpha \hat{\mathbf{p}}. \tag{1.48}$$

For a minimization problem, the best value of the step-size,  $\alpha$ , is the value for which the value of the PES is minimized. Explicitly,

$$\boldsymbol{\alpha}_{\text{optimal}} = \arg\min_{\alpha>0} U\left(\mathbf{x}^{\text{old}} + \alpha \hat{\mathbf{p}}\right)$$
(1.49)

For a transition-state optimization, the energy could increase or decrease, but one can ensure that one is closer to a critical point by minimizing the size of the gradient,

$$\boldsymbol{\alpha}_{\text{optimal}} = \arg \min_{\boldsymbol{\alpha} > 0} \left| \mathbf{g} \left( \mathbf{x}^{\text{old}} + \boldsymbol{\alpha} \hat{\mathbf{p}} \right) \right|^2.$$
(1.50)

Using the optimal value of  $\alpha$  gives a robust, but inefficient, method. The inefficiency arises because determining the optimal step-size requires many evaluations of the PES (Eq. (1.49)) or its gradient (Eq. (1.50)). One can save many evaluations of the PES (or gradient) by only approximately optimizing  $\alpha$  and the robustness of the (quasi-)Newton method is preserved as long as sufficient progress towards the solution is made in each iteration. The mathematical analysis of what constitutes "sufficient progress," that is, how sloppy one can be in the optimization of  $\alpha$  without compromising numerical robustness, was performed by Armijo and Wolfe.<sup>71-73</sup>

Treating only minimization problems, Armijo demonstrated that the step reduced the value of the function enough if  $^{71}$ 

$$U\left(\mathbf{x}^{\text{old}} + \alpha \hat{\mathbf{p}}\right) \leq U\left(\mathbf{x}^{\text{old}}\right) + c_1 \alpha \left(\mathbf{g}^{\text{old}} \cdot \hat{\mathbf{p}}\right)$$
(1.51)

where  $c_1$  is a constant, typically chosen between  $10^{-4}$  and  $10^{-1.49,57}$  This condition is usually supplemented by the Wolfe condition, which forces the slope of the curve  $\phi(\alpha) = U(\mathbf{x}^{\text{old}} + \alpha \hat{\mathbf{p}})$  to decrease,<sup>72,73</sup>

$$\hat{\mathbf{p}} \cdot \mathbf{g} \left( \mathbf{x}^{\text{old}} + \alpha \hat{\mathbf{p}} \right) \ge c_2 \hat{\mathbf{p}} \cdot \mathbf{g}^{\text{old}}, \qquad (1.52)$$

where  $c_2$  is a constant, typically chosen between 0.5 and 0.9.<sup>49,57</sup> In practice, convergence is often better when one imposes the strong Wolfe condition, which forces the magnitude of the slope of  $\phi(\alpha)$  to decrease,<sup>72,73</sup>

$$\left| \hat{\mathbf{p}} \cdot \mathbf{g} \left( \mathbf{x}^{\text{old}} + \alpha \hat{\mathbf{p}} \right) \right| \ge c_2 \left| \hat{\mathbf{p}} \cdot \mathbf{g}^{\text{old}} \right|.$$
(1.53)

For transition-state optimizations, only Eq. (1.53) may be imposed.

#### 2. Trust radius methods

The disadvantage of the line search method is that several expensive evaluations of the PES are made for each (quasi-)Newton step. This seems particular wasteful in quasi-Newton methods: exploration in new directions adds useful information that is used to improve the approximate Hessian, while repeated steps in the same direction does not. The trust-radius method is designed so that there is usually only one evaluation of the PES for each quasi-Newton step.

Trust radius methods are based on the idea that there is a region of radius  $\tau$ , centered at the current point, in which the quadratic model for the PES (cf. Eq. (1.31)) is accurate.<sup>49,57</sup> When solving the quadratic model for the step, as in Eq. (1.34), results in a step that is larger than the trust radius, the step can be scaled back to the trust radius. I.e.,

$$\Delta \mathbf{x} = -\left(\mathbf{H}^{\text{old}}\right)^{-1} \mathbf{g}^{\text{old}} \cdot \min\left(1, \frac{\tau}{\left|\left(\mathbf{H}^{\text{old}}\right)^{-1} \mathbf{g}^{\text{old}}\right|}\right)$$
(1.54)

However, it is better to treat the problem as a constrained optimization,

$$\underbrace{\operatorname{opt}}_{(\Delta \mathbf{x})^{T} \Delta \mathbf{x} \le \tau^{2}} \left( U\left(\mathbf{x}^{\operatorname{old}}\right) + \mathbf{g}^{\operatorname{old}} \cdot \Delta \mathbf{x} + \frac{1}{2} \left(\Delta \mathbf{x}\right)^{T} \mathbf{H}^{\operatorname{old}} \Delta \mathbf{x} \right).$$
(1.55)

When the step predicted by the quadratic model is shorter than  $\tau$ , the constraint is not active. When the unconstrained step is larger than  $\tau$ , the inequality constraint becomes an equality constraint that can be enforced using a Lagrange multiplier,  $\tilde{\lambda}$ . For a positive-semidefinite Hessian, the new equation that needs to be solved is,

$$\left(\mathbf{H}^{\text{old}} + \tilde{\lambda}\mathbf{I}\right)\Delta\mathbf{x} = -\mathbf{g}, \qquad (1.56)$$

with  $\tilde{\lambda} > 0$ . For a negative-semidefinite Hessian,  $\tilde{\lambda} < 0$ . In any case, rewriting the Hessian in terms of its eigenvalues and eigenvectors,

$$\mathbf{H}^{\text{old}} = \sum_{i=1}^{N_{\text{int}}} \lambda_i \boldsymbol{\chi}_i \boldsymbol{\chi}_i^T, \qquad (1.57)$$

a formal solution can be written as:<sup>63,74</sup>{Helgaker, 1991 #6013}

$$\Delta \mathbf{x}\left(\tilde{\lambda}\right) = \sum_{i=1}^{N_{\text{int}}} \frac{-1}{\lambda_i + \tilde{\lambda} \operatorname{sgn}(\lambda_i)} \mathbf{\chi}_i \left(\mathbf{\chi}_i^T \mathbf{g}\right), \qquad (1.58)$$

where  $\tilde{\lambda} > 0$  is determined by solving an equation that forces the step size to be equal to the trust radius,

$$\left(\Delta \mathbf{x}\left(\tilde{\lambda}\right)\right)^{T} \Delta \mathbf{x}\left(\tilde{\lambda}\right) = \tau^{2}$$
(1.59)

In Eq. (1.58) equation,  $sgn(\lambda_i)$  denotes the sign (+1 or -1) of  $\lambda_i$ . This has the effect of shifting negative eigenvalues so that they are more negative, and positive eigenvalues so that they are more positive.

After a step has been taken, the validity of the quadratic model is assessed by comparing the result one would expect if the quadratic model were exact from the result one actually obtains by computing the potential energy and its gradient at the new, updated point. If the quadratic model is accurate, the trust radius is increased; if it is inaccurate, the trust radius is decreased; if the accuracy is moderate, the trust radius remains unmodified.<sup>49,57</sup> Most trust-radius updating schemes are designed primarily for minimization of functions. A new method, more appropriate for transition-state optimization, is presented in Chapter 3.

In addition to the generic trust-radius approach that is presented here, several authors have designed variants specifically for transition-state optimization.<sup>63,74-83</sup> One family of methods, often called eigenvector-following techniques, is based on following a potential-energy surface uphill in the direction of the lowest eigenvalue. The simplest approach requires taking a step in the direction of the lowest eigenvector, and minimizing with respect to all other coordinates.<sup>75,84</sup> A refinement is to consider a step with the form of Eq. (1.58), but to choose  $\tilde{\lambda}$  to be the number between the lowest two eigenvalues of the Hessian,  $\lambda_1 < \tilde{\lambda} < \lambda_2$ , that minimizes the length of the step. (If the resulting step is still too long, it is scaled back linearly by using the modified Hessian in Eq. (1.54).)<sup>76</sup> This basic approach can be refined in numerous ways.<sup>77-79</sup> A particularly powerful revision is to choose separate Lagrange multipliers for the negative-curvature and positive-curvature eigenvectors. This generalizes Eq. (1.58) to

$$\Delta \mathbf{x} = \frac{-1}{\lambda_1 - \lambda_n} \boldsymbol{\chi}_1 \left( \boldsymbol{\chi}_1^T \mathbf{g} \right) + \sum_{i=2}^{N_{int}} \frac{-1}{\lambda_i + \lambda_p} \boldsymbol{\chi}_i \left( \boldsymbol{\chi}_i^T \mathbf{g} \right)$$
(1.60)

The nonnegative Lagrange multipliers,  $\lambda_p$  and  $\lambda_n$ , must be chosen so that the Hessian has the right eigenvector structure. This means, in particular, that one must have  $\lambda_1 - \lambda_n < 0$ and  $\lambda_i + \lambda_p > 0$ .

The most popular ways to choose the Lagrange multipliers are based on the rational function model.<sup>63,68,74,80-83</sup> The idea is that the third- and higher-order terms in the Taylor series expansion of the PES, (1.31), can be mimicked by a Pade approximant,

$$U(\mathbf{q}) \approx U(\mathbf{q}^{\text{old}}) + \frac{\left(\mathbf{g}^{\text{old}}\right)^{T} \left(\mathbf{q} - \mathbf{q}^{\text{old}}\right) + \frac{1}{2} \left(\mathbf{q} - \mathbf{q}^{\text{old}}\right)^{T} \mathbf{H}^{\text{old}} \left(\mathbf{q} - \mathbf{q}^{\text{old}}\right)}{1 + \left(\mathbf{q} - \mathbf{q}^{\text{old}}\right)^{T} \mathbf{S} \left(\mathbf{q} - \mathbf{q}^{\text{old}}\right)}, \quad (1.61)$$

 $S \neq 0$  should be chosen so that this expression is as accurate as possible. In most problems, one has very little information about how one should choose S, so one chooses it to be a constant multiple of the identity matrix,  $S = \zeta I$ , where  $\zeta$  is chosen so that the step size is equal to or less than the trust radius.

The stationary points of the approximate PES in (1.61) are obtained by solving the generalized eigenvalue problem,

$$\begin{bmatrix} \mathbf{H}^{\text{old}} & \mathbf{g}^{\text{old}} \\ \left(\mathbf{g}^{\text{old}}\right)^{T} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x} \\ 1 \end{bmatrix} = 2\left(\Delta E\right) \begin{bmatrix} \mathbf{S} & \mathbf{0} \\ \mathbf{0}^{T} & 1 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x} \\ 1 \end{bmatrix}$$
(1.62)

Taking the lowest eigenvalue gives a step that is appropriate for minimization problems; taking the second lowest eigenvalue/eigenvector gives a step that is appropriate for transition-state finding.

Especially for transition-state searches where the Hessian has more than one negative eigenvalue, or where the negative eigenvalue of the Hessian does not correspond

to the vibrational mode that one believes to be associated with the transition state, it is favorable to consider a more general version of Eq. (1.62) that gives solutions for the Lagrange multipliers in Eq. (1.60). First, one reorders the eigenvalues and eigenvectors of the Hessian so that the one with direction that corresponds to the transition state (typically, the lowest-frequency mode) is listed first. Then one solves two generalized eigenvalue problems for the negative-curvature and positive-curvature Lagrange multipliers,

$$\begin{bmatrix} \lambda_1 & \boldsymbol{\chi}_1^T \mathbf{g} \\ \mathbf{g}^T \boldsymbol{\chi}_1 & \mathbf{0} \end{bmatrix} \mathbf{v}_n = \lambda_n \begin{bmatrix} \zeta & \mathbf{0} \\ \mathbf{0} & \mathbf{1} \end{bmatrix} \mathbf{v}_n$$
(1.63)

$$\begin{bmatrix} \lambda_{2} & 0 & 0 & \cdots & \boldsymbol{\chi}_{2}^{T} \mathbf{g} \\ 0 & \lambda_{3} & 0 & \cdots & \boldsymbol{\chi}_{3}^{T} \mathbf{g} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_{3N-6} & \boldsymbol{\chi}_{3N-6}^{T} \mathbf{g} \\ \mathbf{g}^{T} \boldsymbol{\chi}_{1} & \mathbf{g}^{T} \boldsymbol{\chi}_{2} & \cdots & \mathbf{g}^{T} \boldsymbol{\chi}_{3N-6} & 0 \end{bmatrix} \mathbf{v}_{p} = \lambda_{p} \begin{bmatrix} \zeta & 0 & 0 & \cdots & 0 \\ 0 & \zeta & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & \zeta & 0 \\ 0 & 0 & \cdots & 0 & 1 \end{bmatrix} \mathbf{v}_{p}$$

$$(1.64)$$

One takes the highest eigenvalue of Eq. (1.63) and sets it equal to  $\lambda_n$ ; one takes the lowest eigenvalue of Eq. (1.64) and sets it equal to  $-\lambda_p$ . (In this way both  $\lambda_n$  and  $\lambda_p$  are positive except when the gradient is zero, where they are zero.)

# **1.5** The Initial Geometry Guess for Transition State Optimization

## A. The Problem of Finding Good Initial Guesses

Geometry minimization is relatively straightforward: given a starting point, one uses an algorithm to select downhill steps, inexorably converging to a local minimum on the PES. Transition-state optimization is more complicated because one must maximize the function in one direction, while minimizing it in the others. Good steps could increase or decrease the energy and, if the starting point is near a local minimum on the PES, even the magnitude of the gradient could increase in a good step. It is sometimes difficult to know, beforehand, which direction one should be maximizing in, and which directions one should be minimizing in. For this reason, transition-state optimization is a much less forgiving problem.

In practice, most TS optimizers work well if the initial guess geometry is close enough to the transition state for a quadratic model of the PES to reproduce the qualitative features of the TS region. Specifically, TS optimizers work well when there is one, and only one, negative curvature direction in the initial Hessian, and when this curvature direction corresponds to the transition-state-crossing reaction coordinate. With a good initial guess, most TS optimization methods will successfully converge. With a poor initial guess, most TS optimization methods are unreliable. This underscores the importance of good initial guesses for TS structures.

There are three main approaches to guessing transition-state structures. The most common approach is to guess a transition-state structure based on chemical intuition, typically gleaned from past experience with similar reactions. In the hands of an experienced researcher, this approach is often uncannily effective, but it cannot be automated.

The other approaches search for the transition state starting from the molecular structures of the reactant and/or product.<sup>2,9</sup> Single-ended methods exploit either the reactant or the product, and can be used to explore reactions where the product (or reactant) is unknown. Double-ended methods use both the reactant and the product and are usually more reliable, but they are relatively computationally demanding.

#### **B.** Single-Ended Methods

Single-ended methods start from a single point on the PES and build a path, pointby-point, that leads to a transition-state estimate. Usually the starting point for the path is a reactant or product structure, but this is not essential.

The simplest single-ended method is coordinate driving: an internal coordinate that is believed to characterize a reaction is slowly increased from its value at the starting point. This defines a one-dimensional reduced potential energy surface,

$$u(q_{\rm red}) = \min_{q_2, q_3, \stackrel{\sim}{\longrightarrow} q_n} U(q_{\rm red}, q_2, q_3, \stackrel{\sim}{\longrightarrow}, q_n), \tag{1.65}$$

in which the value of the driven coordinate is fixed and the value of all other coordinates is optimized. The highest point on the reduced PES is taken as a guess for the transition state. This approach is simple and effective when a single coordinate characterizes the reaction pathway, but it often fails when this is not true, which occurs when the reaction path is curved. Mathematically, finding an appropriate choice of coordinate always exists, but finding it often requires creativity. For example, in the isomerization of HCN to HNC, a good choice for the coordinate is the HNC angle, which changes from  $0^{\circ}$  to  $180^{\circ}$  during the course of the reaction. For atom-transfer reactions,

$$D - T \cdots A \to D \cdots T - A, \qquad (1.66)$$

a good choice for the coordinate is often the differential bond length of the transferred atom to the donor and the acceptor,  $q_{red} = |\mathbf{R}_A - \mathbf{R}_T| - |\mathbf{R}_D - \mathbf{R}_T|$ . The telltale signs of failures in the coordinate-driving methods are discontinuous changes in molecular geometry and/or energy.<sup>85-88</sup> This often occurs when there are molecular torsions that accompany a chemical reactions. Other failures occur when the coordinate that is driven does not increase (decrease) monotonically during the reaction.

Failures of the coordinate-driving method are invariably associated with poor choices for the coordinate to be driven. For reactions involving large molecules, such failures are so prevalent as to be nearly inevitable. This has led to a number of heuristic approaches for choosing (often very complicated) coordinates. The simplest approach is to follow the direction-of-least-ascent.<sup>75,84</sup> This approach works well for small molecules and simple reactions, but for larger molecules, all of the most slowly ascending directions tend to correspond to conformational changes (e.g., bond torsions) rather than chemical reactions. For larger molecules, it is favorable to choose an eigenvector of the Hessian that corresponds to the reaction-barrier-crossing motion, and then follow that eigenvector uphill to the transition state. This is the essence of the simple eigenvector-following transition-state finding methods previewed in the last section,<sup>76,77</sup> and their later refinements.<sup>63,68,74,80-83</sup> The dimer method<sup>69</sup> and its refinements<sup>68</sup> are closely related to the

simple eigenvector-following approach, with the feature that it uses gradient calculations at two closely-spaced points on the PES to approximate the Hessian.

Another approach is to construct a path that is guaranteed to pass through stationary points on the potential energy surface. One such path is the gradient-extremal path.<sup>89-96</sup> At a stationary point on a potential energy surface, the gradient is zero and so the gradient is an eigenvector of the Hessian. Therefore, starting at any stationary point, one can define a pathway that will lead to another stationary point by specifying that at every point on the path, the gradient is an eigenvector of the Hessian,  $Hg = \lambda g$ . One hopes that this path will eventually lead to the transition state of interest, though if it does so, it often does so by a circuitous approach.<sup>94</sup> Different gradient extremal paths (corresponding to different eigenvalues of the Hessian) always intersect at stationary points at the PES, so intersections of gradient extremal paths are necessary, but not sufficient, conditions for stationary points. Like eigenvector following techniques, the direct implementation of this approach requires a computationally demanding computation of the Hessian at each step.<sup>93,95,96</sup>

The need for Hessian information is avoided when one considers reducedgradient-following methods. In this approach, one defines a Newton trajectory by specifying that the gradient of the PES always points in the same direction. Like gradient extremal paths, different Newton trajectories intersect at stationary points.<sup>92,97-102</sup> The reduced gradient following approach is usually implemented using the growing string method,<sup>103-105</sup> though more efficient methods have recently been proposed.<sup>102</sup>

38

The fundamental weakness of all these approaches is that one is searching for performing a one-dimensional search for the transition-state structure; this is very effective for small molecules and model systems, but rarely reliable for complicated reactions where the PES can have an enormous number of dimensions. This suggests that one search in several dimensions at once. I.e., one should define a reduced set of coordinates (which are believed to be the key coordinates) and minimize with respect to the remaining coordinates, defining a reduced potential energy surface,

$$u(q_1, q_2, \underline{\rightarrow}, q_r) = \min_{q_{r+1}, q_{r+2}, \underline{\rightarrow}, q_n} U(q_1, q_2, \underline{\rightarrow}, q_r, q_{r+1}, q_{r+2}, \underline{\rightarrow}, q_n).$$
(1.67)

Surface-walking methods search for transition states on this higher-dimensional surface.<sup>106-109</sup> The surface-walking method of Bofill<sup>107</sup> and the scaled-hypersphere method of Ohno and Maeda are multidimensional analogues of the coordinate-driving approach.<sup>110-113</sup> The approach of Irikura and Johnson is analogous to the slowest-ascent approach.<sup>106</sup> The fast-marching method is a wavefront propagation method that leads to an efficient and unified implementation for many surface-walking methods, including the slowest-ascent and geodesic (coordinate-driving) searches.<sup>18,109,114-117</sup> An advantage of the fast-marching approach is that it not only locates stable structures on the PES, but also locates a reaction pathway between the starting structure and any other structure of interest.

Surface walking methods are robust if the set of reduced coordinates is large enough, but they are also prohibitively expensive because their computational cost grows exponentially with the dimensionality of the reduced PES. Expanding beyond five or six dimensions is computationally prohibitive, and since the reduced PES captures only a tiny fraction of the full dimensionality of the full molecular PES, these methods still fail when important coordinates are omitted. However, in cases where the most likely key coordinates can be identified, surface walking methods tend to be robust, but slow.

#### **C.** Double-Ended Methods

Double-ended methods work by bracketing the transition state between the reactant and product structures. This works well because usually (but not always) many of the bond lengths and angles have values in the transition-state structure that are in between their values in the reactant an product structures.

There are two main strategies for double-ended methods. In the first strategy, a pathway connecting the reactant and product is proposed, and the energy is (approximately) maximized along that pathway. The energy-maximizing structure is then taken as a guess for the transition state. The linear synchronous transit (LST)<sup>118</sup> and quadratic synchronous transit (QST)<sup>67</sup> methods use linear and parabolic paths to connect the reactant and product structures. (These paths are typically expressed in terms of interatomic distance or internal coordinates; Cartesian coordinates are much less effective.) In the QST3 method, a guess for the transition state is used to obtain a better parabolic path.

The first disadvantage of these methods is that they are relatively expensive, since several energy/gradient calculations along the path are required. To overcome this problem, a new approach that does not require evaluating the PES, but which is still extremely effective, is presented in chapter 2. The larger problem is that the pathway is highly approximate: the true minimum-energy reaction path or (very similar, and usually identical) intrinsic reaction coordinate is usually very different from the guessed pathway.<sup>19,119 120</sup>

To obtain a good approximation to the minimum-energy reaction path, it is necessary to optimize the pathway. Most approaches for doing this are chains-of-states models: the energy/gradient/Hessian is evaluated (or approximated) at a series of points, and the reaction path is a curve that interpolates between these points. The lowest-energy pathway is then obtained by minimizing the energy of the points along the path. If one minimized the points without constraints, then all the points would fall into the nearest local minimum. It is therefore essential to add appropriate constraints to the minimization so that the points remain relatively evenly spaced along the pathway.

In the nudged elastic band (NEB) methods,<sup>121-129</sup> one imagines the reaction path as a series of points (where the PES and its gradient is evaluated) connected by springs, with potential

$$V^{\text{spring}}\left(\mathbf{x}_{i}\right) = \frac{1}{2}k\left(\left|\mathbf{x}_{i} - \mathbf{x}_{i-1}\right|^{2} + \left|\mathbf{x}_{i} - \mathbf{x}_{i+1}\right|^{2}\right)$$
(1.68)

The springs keep the points nearly equally spaced. To ensure that the NEB path goes through the transition state, one should remove the component of the restoring force that is normal to the reaction path. The resulting method uses the gradient of the potential energy surface to guide motions perpendicular to the reaction path, and the gradient of the spring to guide motions parallel to the path,

$$\mathbf{g}_{i}^{\text{NEB}} = \left[\mathbf{g}_{i}^{\text{PES}} - \left(\mathbf{g}_{i}^{\text{PES}} \cdot \mathbf{\tau}_{i}\right)\mathbf{\tau}_{i}\right] + \left[\left(\mathbf{g}_{i}^{\text{spring}} \cdot \mathbf{\tau}_{i}\right)\mathbf{\tau}_{i}\right]$$
(1.69)

where  $\tau_i$  is the tangent to the path at  $\mathbf{x}_i$  and  $\mathbf{g}^{\text{PES}}$  and  $\mathbf{g}^{\text{spring}}$  are the gradients of the PES and the spring potential, respectively.

String methods imagine the reaction path as a necklace of points threaded by a string, which is typically a cubic spline.<sup>130-135</sup> The steepest-descent path connecting the transition state to the reactant and the product follows a gradient-descent line; therefore the tangent line to the path is parallel to the gradient,

$$\mathbf{\tau}_{i} = \frac{d\mathbf{x}(t)}{dt} \propto \mathbf{g}.$$
(1.70)

For an approximate path, component of the gradient perpendicular to the path,

$$\mathbf{g}_{\perp} \equiv \mathbf{g}_{i}^{\text{PES}} - \left(\mathbf{g}_{i}^{\text{PES}} \cdot \mathbf{\tau}_{i}\right) \mathbf{\tau}_{i}$$
(1.71)

is not zero. String methods work by updating the path to minimize  $|\mathbf{g}_{\perp}|^2$ . One of the most efficient string methods is the quadratic string method (QSM), which constructs an approximate quadratic model for minimizing  $|\mathbf{g}_{\perp}|^2$ .<sup>134</sup>

Because there is no fictitious potential maintaining the spacing between the points along the string, the evaluation-points in the string methods are less coupled together than the points in nudged elastic band approaches, enhancing convergence. A problem with the string methods is that the points along the path need to be periodically redistributed to ensure that the transition-state region is adequately sampled. In addition, if the steps are too large, the string can become kinked, and artificial forces must be added to the potential to straighten out the path. If the steps are too small, convergence is unnecessarily slow. Therefore, while string methods converge more quickly than nudged elastic band approaches, they also require more parameter-tuning. Closely related to string methods are splined-saddle approaches.<sup>136,137</sup> In splinedsaddle methods, the ends of the string are not the reactant and the product, but intermediate structures that are closer to the transition state. By reducing the length of the string, this increases the computation speed with which a transition-state guess may be obtained. However, unlike the other double-ended methods presented in this section, splined-saddle methods do not guarantee that the located transition state connects the reactant and product structures.

The sequential quadratic programming method proposed by Burger and Yang is similar in spirit to string methods, but simpler to implement.<sup>133</sup> In it, each point in an initial guess for the reaction path is minimized on the hyperplane perpendicular to the path using a local quadratic model for the potential energy surface. This approach works well for simple reactions but, like the string method by which it was inspired, requires periodic redistribution of points and must the monitored to ensure that it does not converge to a kinked path.

## **1.6 Summary**

Building computational models for chemical reactions is among the most fundamental and important problems in theoretical chemistry. There are several aspects to this problem, from accurate determination of the molecular energy to geometry optimization: locating the stationary points on the potential-energy surface that characterize the stable molecular structures (reactants, products, and reactive intermediates) and the transition-states that connect them.

This chapter reviews recent work on geometry optimization, emphasizing the difficult problem of transition-state optimization and featuring our recent work on this topic. The key ingredients of good geometry optimization methods are discussed. The starting point of any optimization problem is a good starting structure. It is usually relatively easy to find good starting structures for minimization to a stable reactant/product/intermediate, but finding good starting structures for transition-state optimization is more challenging. Some methods for estimating transition states are reviewed in section 1.5. In chapter 2, we present a new, extremely computationally efficient, approach based on interpolating between reactant and product structures in redundant internal coordinates. After an initial structure has been obtained, one needs to select a coordinate system in which to perform the optimization. In most cases, redundant internal coordinates is the best choice, and the mathematical intricacies of this approach are reviewed in section 1.3. A fuller account of this approach, including a discussion of the robust dihedral coordinates from Eqs. (1.29)-(1.30), is presented in chapter 2.

Most geometry minimization and transition-state optimization methods are based on Newton's method (in the rare cases where it is affordable to compute the Hessian matrix exactly) or quasi-Newton approaches (in which the Hessian is approximated using the gradients computed in successive steps of the optimization). These methods are described in section 1.4. Chapter 3 presents a new hybrid approach, where the elements of the Hessian matrix associated with key chemical changes are explicitly evaluated (by finite difference) and the remaining elements of the Hessian are approximated with quasi-Newton updates. This approach is tested in Chapter 4, which shows that the method is nearly as efficient as the best transition-state optimizers but, importantly, is significantly more likely to converge. This new approach is decisively better than conventional approaches when the initial guess for the transition-state structure is poor.

Another alternative approach is to combine the minimization of the electronic degrees of freedom with the minimization of the nuclear degrees of freedom.<sup>138-141</sup> Such an approach can be more efficient than the gradient-based approaches usually considered (in which the electronic degrees of freedom and nuclear degrees of freedom are optimized sequentially). Applying that approach to quantum chemistry methods (Møller-Plesset perturbation theory, coupled-cluster methods, etc.) that are not variational seems quite complicated. (It can be done by viewing the geometry optimization as a system of nonlinear equations, in which the gradient of the energy with respect to the nuclear coordinates is zero. This system of nonlinear equations can then be augmented by the nonlinear equations determining the energy in the quantum chemistry method.) In practice, the increase in efficiency obtained by simultaneous optimization of electronic and nuclear degrees of freedom seems relatively modest, and those techniques are less adaptable to transition-state-finding and nonvariational quantum chemistry methods. The robustness and flexibility of gradient-based optimization methods explains why that approach is usually preferred.

As emphasized in this chapter, the performance of a transition-state optimizer depends on many factors. Chapter 5 shows how the choice of initial Hessian can
influence the performance of transition-state finders, and shows that a minimal-basis-set Hartree-Fock calculation typically provides an excellent starting point for transition-state optimizations. Chapter 6 shows how the method presented in Chapter 3 can be transformed into a true black-box transition-state finder by using methods to automatically select the chemically-important internal coordinates. Chapter 7 discusses a few other key parameters, including the type and selection of dihedral angles, the trustradius method, the quasi-Newton update, etc. Finding the *best* transition-state finder is quixotic, but by making good algorithmic choices, robust and rapid transition-state optimization methods can be designed.

## **1.7 References**

- (1) Schlegel, H. B. J. Comput. Chem. 2003, 24, 1514.
- (2) Wales, D. J. *Energy landscapes*; Cambridge University Press: Cambridge,

UK ; New York, 2003.

- (3) McQuarrie, D. A.; Harper-Collins: New York, 1976.
- (4) Peng, C. Y.; Ayala, P. Y.; Schlegel, H. B.; Frisch, M. J. J. Comput. Chem.

**1996**, *17*, 49.

- (5) Bakken, V.; Helgaker, T. J. Chem. Phys. 2002, 117, 9160.
- (6) Pulay, P.; Fogarasi, G. J. Chem. Phys. 1992, 96, 2856.
- (7) Eckert, F.; Pulay, P.; Werner, H. J. J. Comput. Chem. **1997**, *18*, 1473.
- (8) Farkas, O.; Schlegel, H. B. J. Mol. Struct.: THEOCHEM 2003, 666, 31.
- (9) Schlegel, H. B. Wiley Interdisciplinary Reviews-Computational Molecular Science **2011**, *1*, 790.

(10) Levine, I. N. *Quantum chemistry*; 6th ed.; Pearson Prentice Hall: Upper Saddle River, N.J., 2009.

(11) Helgaker, T.; Jørgensen, P.; Olsen, J. Modern electronic structure theory;Wiley: Chichester, 2000.

(12) Simons, J.; Nichols, J. *Quantum mechanics in chemistry*; Oxford UP: New York, 1997.

(13) Cramer, C. J. *Essentials of computational chemistry*; Wiley: Chichester, 2002.

(14) Head-Gordon, M. J. Phys. Chem. 1996, 100, 13213.

(15) Raghavachari, K.; Anderson, J. B. J. Phys. Chem. 1996, 100, 12960.

(16) Wales, D. J.; Doye, J. P. K.; Miller, M. A.; Mortenson, P. N.; Walsh, T. R.
In Advances in Chemical Physics, Volume 115; Prigogine, I. R. S. A., Ed. 2000; Vol. 115, p 1.

(17) Wilson, E. B.; Cross, P. C.; Decius, J. C. *Molecular vibrations : the theory of infrared and Raman vibrational spectra*; Dover Publications: New York, 1980.

(18) Dey, B. K.; Janicki, M. R.; Ayers, P. W. J. Chem. Phys. 2004, 121, 6667.

(19) Burger, S. K., Duke University, 2007.

(20) Gear, C. W. Numerical initial value problems in ordinary differential equations; Prentice-Hall: Englewood Cliffs, N.J., 1971.

(21) Garrett, B. C.; Redmon, M. J.; Steckler, R.; Truhlar, D. G.; Baldridge, K.

K.; Bartol, D.; Schmidt, M. W.; Gordon, M. S. J. Phys. Chem. 1988, 92, 1476.

- (22) Ishida, K.; Morokuma, K.; Komornicki, A. J. Chem. Phys. 1977, 66, 2153.
- (23) Page, M.; Doubleday, C.; McIver, J. W. J. Chem. Phys. 1990, 93, 5634.
- (24) Sun, J. Q.; Ruedenberg, K. J. Chem. Phys. 1993, 99, 5257.
- (25) Gonzalez, C.; Schlegel, H. B. J. Chem. Phys. 1991, 95, 5853.
- (26) Gonzalez, C.; Schlegel, H. B. J. Phys. Chem. 1990, 94, 5523.
- (27) Burger, S. K.; Yang, W. T. J. Chem. Phys. 2006, 125, 244108.
- (28) Burger, S. K.; Yang, W. T. J. Chem. Phys. 2006, 124, 224102.
- (29) Baker, J.; Chan, F. R. J. Comput. Chem. 1996, 17, 888.

(30) Fogarasi, G.; Zhou, X. F.; Taylor, P. W.; Pulay, P. J. Am. Chem. Soc.1992, 114, 8191.

- (31) Baker, J.; Kessi, A.; Delley, B. J. Chem. Phys. 1996, 105, 192.
- (32) Baker, J.; Pulay, P. J. Chem. Phys. 1996, 105, 11100.
- (33) Baker, J.; Kinghorn, D.; Pulay, P. J. Chem. Phys. 1999, 110, 4986.
- (34) Baker, J.; Pulay, P. J. Comput. Chem. 2000, 21, 69.
- (35) von Arnim, M.; Ahlrichs, R. J. Chem. Phys. 1999, 111, 9183.
- (36) Farkas, O.; Schlegel, H. B. J. Chem. Phys. 1998, 109, 7100.
- (37) Billeter, S. R.; Turner, A. J.; Thiel, W. PCCP 2000, 2, 2177.
- (38) Paizs, B.; Baker, J.; Suhai, S.; Pulay, P. J. Chem. Phys. 2000, 113, 6566.
- (39) Kudin, K. N.; Scuseria, G. E.; Schlegel, H. B. J. Chem. Phys. 2001, 114,

2919.

- (40) Bucko, T.; Hafner, J.; Angyan, J. G. J. Chem. Phys. 2005, 122.
- (41) Doll, K.; Dovesi, R.; Orlando, R. Theor. Chem. Acc. 2006, 115, 354.
- (42) Gordon, M. S.; Pople, J. A. J. Chem. Phys. 1968, 49, 4643.
- (43) Baker, J.; Hehre, W. J. J. Comput. Chem. 1991, 12, 606.
- (44) Baker, J. J. Comput. Chem. 1993, 14, 1085.
- (45) Pulay, P.; Fogarasi, G.; Pang, F.; Boggs, J. E. J. Am. Chem. Soc. 1979,

101, 2550.

(46) Bouaricha, A. ACM Transactions on Mathematical Software 1997, 23, 81.

(47) Fletcher, R. *Practical methods of optimization*; 2nd ed.; Wiley: Chichester; New York, 1987.

(48) Gill, P. E.; Murray, W.; Wright, M. H. *Practical optimization*; Academic Press: London ; Toronto, 1981.

(49) Dennis, J. E.; Schnabel, R. B. *Numerical methods for unconstrained optimization and nonlinear equations*; Prentice-Hall: Englewood Cliffs, N.J., 1983.

(50) Scales, L. E. *Introduction to non-linear optimization*; Macmillan: London, 1985.

(51) Nocedal, J.; Wright, S. J. In Springer series in operations research; Springer: New York, 1999, p xx.

(52) Thiel, W. In Advances in Chemical Physics, Vol Xciii: New Methods in Computational Quantum Mechanics; Prigogine, I., Rice, S. A., Eds. 1996; Vol. 93, p 703.

(53) Bredow, T.; Jug, K. Theor. Chem. Acc. 2005, 113, 1.

(54) Schlegel, H. B. *Theor. Chim. Act.* **1984**, *66*, 333.

- (55) Wittbrodt, J. M.; Schlegel, H. B. J. Mol. Struct.: THEOCHEM 1997, 398,
- 55.

(56) Fischer, T. H.; Almlof, J. J. Phys. Chem. 1992, 96, 9768.

(57) Nocedal, J.; Wright, S. J. *Numerical Optimization*; Springer-Verlag: New York, 1999.

(58) Fletcher, R. Computer Journal **1970**, 13, 317.

(59) Goldfarb, D. *Mathematics of Computation* **1970**, *24*, 23.

(60) Broyden, C. G. Journal of the Institute of Mathematics and Its Applications **1970**, *6*, 76.

- (61) Shanno, D. F. *Mathematics of Computation* **1970**, *24*, 647.
- (62) Shanno, D. F.; Kettler, P. C. Mathematics of Computation 1970, 24, 657.
- (63) Bofill, J. M. J. Comput. Chem. 1994, 15, 1.

- (64) Farkas, O.; Schlegel, H. B. J. Chem. Phys. 1999, 111, 10806.
- (65) Bofill, J. M. Int. J. Quantum Chem. 2003, 94, 324.
- (66) Burger, S. K.; Ayers, P. W. J. Chem. Phys. 2010, 132, 234110.
- (67) Peng, C. Y.; Schlegel, H. B. Isr. J. Chem. 1993, 33, 449.
- (68) Heyden, A.; Bell, A. T.; Keil, F. J. J. Chem. Phys. 2005, 123.
- (69) Henkelman, G.; Jonsson, H. J. Chem. Phys. 1999, 111, 7010.
- (70) Powell, M. J. D. In Nonlinear Programming 3; Mangasarian, O., Meyer,

M., Robinson, S., Eds.; Academic Press: New York, 1978, p 27.

- (71) Armijo, L. Pacific Journal of Mathematics 1966, 16, 1.
- (72) Wolfe, P. SIAM Review **1969**, *11*, 226.
- (73) Wolfe, P. SIAM Review **1971**, *13*, 185.
- (74) Culot, P.; Dive, G.; Nguyen, V. H.; Ghuysen, J. M. Theor. Chim. Act.

**1992**, *82*, 189.

- (75) Crippen, G. M.; Scheraga, H. A. Arch. Biochem. Biophys. 1971, 144, 462.
- (76) Cerjan, C. J.; Miller, W. H. J. Chem. Phys. 1981, 75, 2800.
- (77) Simons, J.; Jorgensen, P.; Taylor, H.; Ozment, J. J. Phys. Chem. 1983, 87,

2745.

(78) Nichols, J.; Taylor, H.; Schmidt, P.; Simons, J. J. Chem. Phys. 1990, 92,

340.

- (79) Simons, J.; Nichols, J. Int. J. Quantum Chem. 1990, 263.
- (80) Banerjee, A.; Adams, N.; Simons, J.; Shepard, R. J. Phys. Chem. 1985, 89,

52.

- (81) Baker, J. J. Comput. Chem. 1986, 7, 385.
- (82) Besalu, E.; Bofill, J. M. Theor. Chem. Acc. 1998, 100, 265.
- (83) Anglada, J. M.; Bofill, J. M. Int. J. Quantum Chem. 1997, 62, 153.
- (84) Poppinger, D. Chem. Phys. Lett. 1975, 35, 550.
- (85) Burkert, U.; Allinger, N. L. J. Comput. Chem. 1982, 3, 40.

(86) Williams, I. H.; Maggiora, G. M. Theochem-Journal of Molecular Structure 1982, 6, 365.

- (87) Scharfenberg, P. Chem. Phys. Lett. 1981, 79, 115.
- (88) Rothman, M. J.; Lohr, L. L. Chem. Phys. Lett. 1980, 70, 405.
- (89) Pancir, J. Collect. Czech. Chem. Commun. 1975, 40, 1112.
- (90) Basilevsky, M. V.; Shamov, A. G. Chem. Phys. 1981, 60, 347.
- (91) Hoffman, D. K.; Nord, R. S.; Ruedenberg, K. Theor. Chim. Act. 1986, 69,

265.

(92) Bofill, J. M.; Quapp, W.; Caballero, M. J. Chem. Theory Comp. 2012, 8,

927.

(93) Jorgensen, P.; Jensen, H. J. A.; Helgaker, T. Theor. Chim. Act. 1988, 73,

55.

- (94) Sun, J. Q.; Ruedenberg, K. J. Chem. Phys. 1993, 98, 9707.
- (95) Schlegel, H. B. Theor. Chim. Act. 1992, 83, 15.
- (96) Quapp, W. Theor. Chim. Act. 1989, 75, 447.
- (97) Quapp, W. J. Math. Chem. 2004, 36, 365.
- (98) Hirsch, M.; Quapp, W. J. Comput. Chem. 2002, 23, 887.

(99) Quapp, W.; Hirsch, M.; Imig, O.; Heidrich, D. J. Comput. Chem. 1998, 19, 1087.

(100) Crehuet, R.; Bofill, J. M.; Anglada, J. M. Theor. Chem. Acc. 2002, 107,

130.

(101) Bofill, J. M.; Quapp, W. J. Chem. Phys. 2011, 134, 074101.

(102) Liu, Y. L.; Burger, S. K.; Ayers, P. W. J. Math. Chem. 2011, 49, 1915.

(103) Peters, B.; Heyden, A.; Bell, A. T.; Chakraborty, A. J. Chem. Phys. 2004, 120, 7877.

(104) Quapp, W. Journal of Theoretical & Computational Chemistry 2009, 8,101.

(105) Quapp, W. J. Chem. Phys. 2005, 122, 174106.

(106) Irikura, K. K.; Johnson, R. D. J. Phys. Chem. A 2000, 104, 2191.

(107) Bofill, J. M.; Anglada, J. M. Theor. Chem. Acc. 2001, 105, 463.

(108) Anglada, J. M.; Besalu, E.; Bofill, J. M.; Crehuet, R. J. Comput. Chem. **2001**, 22, 387.

- (109) Burger, S. K.; Ayers, P. W. J. Chem. Theory Comp. 2010, 6, 1490.
- (110) Ohno, K.; Maeda, S. Phys. Scr. 2008, 78, 058122.
- (111) Maeda, S.; Watanabe, Y.; Ohno, K. Chem. Phys. Lett. 2005, 414, 265.
- (112) Maeda, S.; Ohno, K. J. Phys. Chem. A 2005, 109, 5742.
- (113) Ohno, K.; Maeda, S. Chem. Phys. Lett. 2004, 384, 277.
- (114) Liu, Y.; Ayers, P. W. J. Math. Chem. 2011, 49, 1291.

(115) Liu, Y. L.; Burger, S. K.; Dey, B. K.; Sarkar, U.; Janicki, M.; Ayers, P. W.

In Quantum Biochemistry; Matta, C. F., Ed.; Wiley-VCH: Boston, 2010.

(116) Burger, S. K.; Liu, Y. L.; Sarkar, U.; Ayers, P. W. J. Chem. Phys. 2009, 130, 024103.

- (117) Dey, B. K.; Ayers, P. W. Mol. Phys. 2006, 104, 541.
- (118) Halgren, T. A.; Lipscomb, W. N. Chem. Phys. Lett. 1977, 49, 225.
- (119) Fukui, K. Acc. Chem. Res. 1981, 14, 363.
- (120) Quapp, W.; Heidrich, D. Theor. Chim. Act. 1984, 66, 245.
- (121) Henkelman, G.; Uberuaga, B. P.; Jonsson, H. J. Chem. Phys. 2000, 113,

9901.

- (122) Henkelman, G.; Jonsson, H. J. Chem. Phys. 2000, 113, 9978.
- (123) Trygubenko, S. A.; Wales, D. J. J. Chem. Phys. 2004, 120, 2082.
- (124) Chu, J. W.; Trout, B. L.; Brooks, B. R. J. Chem. Phys. 2003, 119, 12708.
- (125) Maragakis, P.; Andreev, S. A.; Brumer, Y.; Reichman, D. R.; Kaxiras, E.

J. Chem. Phys. 2002, 117, 4651.

(126) Galvan, I. F.; Field, M. J. J. Comput. Chem. 2008, 29, 139.

(127) Sheppard, D.; Terrell, R.; Henkelman, G. J. Chem. Phys. 2008, 128, 134106.

- (128) Alfonso, D. R.; Jordan, K. D. J. Comput. Chem. 2003, 24, 990.
- (129) Gonzalez-Garcia, N.; Pu, J. Z.; Gonzalez-Lafont, A.; Lluch, J. M.; Truhlar,
- D. G. J. Chem. Theory Comp. 2006, 2, 895.

(130) E, W. N.; Ren, W. Q.; Vanden-Eijnden, E. J. Chem. Phys. 2007, 126, 164103.

- (131) Weinan, E.; Ren, W. Q.; Vanden-Eijnden, E. Phys. Rev. B 2002, 66.
- (132) Ayala, P. Y.; Schlegel, H. B. J. Chem. Phys. 1997, 107, 375.
- (133) Burger, S. K.; Yang, W. J. Chem. Phys. 2007, 127, 164107.
- (134) Burger, S. K.; Yang, W. T. J. Chem. Phys. 2006, 124, 054109.
- (135) Koslover, E. F.; Wales, D. J. J. Chem. Phys. 2007, 127.
- (136) Ghasemi, S. A.; Goedecker, S. J. Chem. Phys. 2011, 135.
- (137) Granot, R.; Baer, R. J. Chem. Phys. 2008, 128.
- (138) Headgordon, M.; Pople, J. A. J. Phys. Chem. 1988, 92, 3063.
- (139) Headgordon, M.; Pople, J. A.; Frisch, M. J. Int. J. Quantum Chem. 1989,

291.

- (140) Marks, L. D. J. Chem. Theory Comp. 2013, 9, 2786.
- (141) Moss, C. L.; Li, X. S. J. Chem. Phys. 2008, 129.

Chapter 2

# **Generating Initial Guesses for Transition-State**

# **Optimization by Interpolating in**

## **Robust Redundant Internal Coordinates**

## 2.1 Overview

Three new methods for guessing an initial geometry for a transition-state optimization, requiring only the reactant and product geometries, are presented and tested for a diverse set of 130 reactions. The new approaches are based on interpolation between the redundant internal coordinates of the reactant and the product; most of the initial geometries have one imaginary frequency (i.e., they lie in the transition-state region) and standard transition-state finding algorithms were able to converge 80% of the reactions in twenty or fewer steps. We circumvent the linear-dihedral problem by introducing robust redundant internal coordinates. In robust redundant internal coordinates, each dihedral angle (bond torsion) is replaced by two new coordinates. Robust redundant internal coordinates are numerically sound even when the rotating atom is nearly collinear with the bond being rotated around.

### 2.2 Motivation

The ability to locate transition states on a molecular potential energy surface is the *sine qua non* of computational chemical kinetics.<sup>1-4</sup> Once the transition state has been located, the rate can be modelled using transition-state theory or, more accurately, by one of the many approaches based on molecular dynamics or Monte Carlo sampling. All of these approaches rely upon a reasonable guess for the transition state, either as an initial guess for a geometry optimization (in the conventional quantum chemistry approach

based on transition-state theory<sup>5</sup>) or to develop a biasing strategy (in conventional statistical mechanics approaches based on sampling reactive trajectories<sup>6-8</sup>). This paper proposes, and tests, a very simple and computationally inexpensive strategy for guessing the transition state based on interpolation, in internal coordinates, between the reactant and product structures. Along the way to this objective, we develop a database of 130 reactions for testing transition-state methods and propose a new way to resolve the ill-conditioning associated with internal coordinates, especially the problems of (a) converting from redundant internal coordinates to Cartesian coordinates and (b) ill-conditioning due to dihedral angles,  $\alpha\beta\gamma\delta$ , in which one of the terminal bond angles,  $\alpha\beta\gamma$ 

or  $\beta\gamma\delta$ , is nearly linear.

## 2.3 Background

#### A. Methods for Guessing Transition States

Most existing approaches for guessing the transition state structures of elementary chemical reactions fall into one of two categories. Single-ended methods start from the reactant (or product) and search the potential-energy surface (PES) for low-energy reaction pathways.<sup>5,9</sup> Double-ended methods start from the reactant and product structures (and sometimes also a guess for the transition-state geometry), attempt to find a reaction pathway that links them, and use the geometry of the highest-energy-point on that pathway as an estimate for the transition-state structure.

The simplest single-ended method is coordinate driving.<sup>10-13</sup> In coordinate-driving, one simply picks a coordinate whose value is different in the reactants and the products, and starting from either the reactant or product, slowly increments the value of the chosen coordinate, while minimizing the energy with respect to all the other coordinates. Coordinate driving merely requires a series of constrained optimizations along the path, so it is relatively computationally efficient. However, it often finds a transition state that is too high in energy. Moreover, if the selected coordinate is inappropriate, or if the reaction pathway cannot be described by a single geometrical coordinate, the molecular geometry and the energy profile can change discontinuously.<sup>5</sup>

Most other single-ended methods can be viewed as refinements of the coordinatedriving idea. In the reduced gradient following method, one starts at a stable structure and constructs the Newton trajectory: a curve for which the gradient of the PES always points in the same direction. Newton trajectories converge on stationary points.<sup>14-20</sup> The reduced gradient following approach is usually implemented using the growing string method,<sup>21-23</sup> though more efficient methods have recently been proposed.<sup>20</sup> In the eigenvectorfollowing method, one picks an eigenvalue of the Hessian (typically the eigenvector corresponding to the lowest-frequency normal mode of vibration) and moves in that direction.<sup>24-32</sup> Unlike the other approaches, eigenvector following requires repeated evaluations of the Hessian along the pathway, so that the eigenvector-direction can be determined. The dimer method is an approach in the same spirit as eigenvector following, but in which the Hessian is not explicitly constructed.<sup>33,34</sup> Methods based on coordinate-driving, eigenvector-following, and reducedgradient following often fail to find the relevant transition state when there are many lowfrequency normal modes, many of which correspond to low-barrier conformational shifts instead of bond-breaking/forming processes.<sup>35</sup> This can be overcome by using surfacewalking methods, where multiple directions are explored at once.<sup>36,37</sup> Surface-walking methods like the scaled hypersphere search<sup>38-41</sup> and the fast-marching method<sup>37,42-46</sup> are robust if the dimensionality of the search is large enough, but they are quite expensive. When a heuristic is used to reduce the dimensionality of the PES that must be explored, surface-walking methods become less reliable.<sup>21</sup>

Double-ended methods start from an initial guess for the reaction pathway, or at least the portion of the reaction pathway that crosses the transition state. They optimize the entire minimum-energy reaction path, or at least the transition-state portion of the reaction path.<sup>5</sup>

In double-ended methods, a reaction path is formed by interpolating between a sequence of geometric structures stretching from the reactant to the product regions of the PES. Different approaches differ in the method for selecting points are generated and in the type of constrained optimization that is performed when updating the coordinates.<sup>47</sup> The most popular double-ended methods are the string methods<sup>47-52</sup> and the nudged elastic band methods.<sup>53-61</sup> (Many variations of these two methods exist.) Double-ended methods are more robust than single-ended methods because they always find a transition state between the reactant and product structures, though the transition state will not be the lowest-energy transition-state structure unless the initial path is chosen correctly.

Unfortunately, double-ended methods are expensive, requiring the PES and its gradient to be evaluated at every point on the reaction path even when only the transition-state structure is of interest. This is somewhat mitigated by the splined-saddle methods,<sup>62,63</sup> which are string methods where the approximate reaction path does not extend all the way to the reactant and product structures. Unlike conventional double-ended methods, however, splined-saddle methods require a good guess for the location of the transition-state region.

#### **B.** Redundant Internal Coordinates

The computational efficiency of both single-ended and double-ended methods is sensitive to the coordinate system one chooses to specify the positions of the atomic nuclei. Often Cartesian coordinates are used because the implementation of methods is easier in Cartesian coordinates, and quantum chemistry software for determining the PES and its derivatives usually uses Cartesian coordinates. Because Cartesian coordinates are tightly coupled to each other, however, optimizations in Cartesian coordinates are generally less efficient.

It is better, computationally and intuitively, to use internal coordinates. Internal coordinates allow one to specify the molecular geometry using chemically intuitive internuclear (bond) distances, angles between bonds, and torsions around bonds (dihedrals). The coupling between these coordinates is much smaller, so the Hessian is more diagonally dominant in internal coordinates than it is in Cartesian coordinates, which increases the accuracy of quasi-Newton updates for the Hessian.<sup>5</sup>

The problem with internal coordinates is their redundancy: only 3N-6 (3N-5 for linear molecules) internal coordinates are needed to specify the molecular geometry of an N atom molecule. However, including all of the bond-distances, angles between bonds, and torsions around bonds usually gives many more coordinates than required. Arbitrarily deleting coordinates reintroduces the strong coupling between coordinates that internal coordinates were designed to avoid in the first place. For chain-like systems, it can be relatively easy to construct a Z-matrix representation of the molecular geometry that is efficient for geometry optimization. However, for cyclic systems, one of the bonds in the ring cannot be included in the Z-matrix, and the length of the omitted bond is a very complicated function of the other coordinates. Rather than attempt to choose a good set of 3N-6 internal coordinates, then, it is preferable to find a way to use the redundant set of internal coordinates.<sup>64-70</sup> Starting from the Cartesian coordinates of the molecule, it is an exercise in trigonometry to determine the values of the redundant internal coordinates. Converting from redundant internal coordinates to Cartesian coordinates is more challenging, partly because there are inconsistent sets of redundant internal coordinates that do not correspond to any real molecular geometry. Iterative procedures are usually used to convert redundant internal coordinates to Cartesian coordinates.<sup>64-79</sup> When unrealizable values for the redundant internal coordinates are used, the iterative methods fail, and the Cartesian coordinates are either specified by an *ad hoc* rule or the software crashes.

#### C. Preview of Our Results

This paper emerged from our attempts to design a better approach for converting from redundant internal coordinates to Cartesian coordinates. In section 2.4.B, we describe the system of internal coordinates we use. In order to maintain good numerical properties, our internal coordinates do not include dihedral angles. Instead we introduce two new coordinates which fully specify the dihedral angle, but which do not lead to numerical ill-conditioning when the first (or last) three atoms in the dihedral are nearly collinear. In section 2.4.C, we present a method for converting from redundant internal coordinates to Cartesian coordinates. The idea of this method is that the values of the redundant internal coordinates that correspond to realizable molecular geometries define a 3N–6-dimensional manifold in the space of redundant internal coordinates. A nonphysical set of redundant internal coordinates is assigned to the nearest point on the manifold are thus assigned to the redundant internal coordinates.

Section 2.4.D uses this idea to estimate the position of a transition state. Specifically, weighted-averages of the reactant- and product-structures in redundant internal coordinates are shown to give excellent approximations to the transition-state structure. Because this approach uses the structure of the reactants and products, it is properly classified as a double-ended method. However, because the PES is never evaluated (not even for the reactant or product), this method is extremely fast. In section 2.5 we test this method for a set of 130 diverse reactions, proving that it gives excellent results.

## 2.4 Methods

### A. Redundant Internal Coordinates

An *N*-atom molecule can be described with 3*N* Cartesian coordinates,  $\{x_i\}_{i=1}^{3N}$ , or with (usually redundant) internal coordinates like interatomic distances, bond angles, and dihedrals,  $\{q_i\}_{i=1}^{M_{int}}$ . The individual internal coordinates are functions of the Cartesian coordinates,  $q_i(\mathbf{x})$ , and the Jacobian of the transformation from Cartesian to internal coordinates is the Wilson **B** matrix, with elements,<sup>65,70,80,81</sup>

$$b_{ij} = \frac{\partial q_i}{\partial x_j} \tag{2.1}$$

The Wilson matrix is generally rectangular, with  $M_{int} = \dim(\mathbf{q}) \ge 3N - 6$  rows and  $3N = \dim(\mathbf{x})$  columns. Usually the number of internal coordinates is several times larger than the number of Cartesian coordinates. By definition, the change in internal coordinates induced by a small change in the Cartesian coordinates can be computed as,

$$\delta \mathbf{q} = \mathbf{B} \,\delta \mathbf{x} \tag{2.2}$$

Since the **B** matrix is rarely square and since it is always singular (because rotation and translation of the molecule changes the Cartesian coordinates, but not the internal coordinates), the inverse of **B** does not exist. The inverse transformation is defined instead as,

$$\delta \mathbf{x} = \mathbf{B}^+ \delta \mathbf{q} \tag{2.3}$$

where  $\mathbf{B}^+$  is the (Moore-Penrose) pseudoinverse of **B**. To compute Cartesian coordinates corresponding to a given set of internal coordinates, one usually starts with a matchedpair of Cartesian and internal coordinates,  $\mathbf{x}_0(\mathbf{q}_0)$ , and then uses Eq. (2.3) to estimate the change in Cartesian coordinates,  $\mathbf{x}_1 = \mathbf{x}_0 + \mathbf{B}^+(\mathbf{q}-\mathbf{q}_0)$ . If the internal coordinates,  $\mathbf{q}_1$ , corresponding to  $\mathbf{x}_1$  are not close enough to the target internal coordinates, then a correction is computed,  $\mathbf{x}_2 = \mathbf{x}_1 + \mathbf{B}^+(\mathbf{q}-\mathbf{q}_1)$ . This procedure is repeated until convergence is achieved. Unfortunately, this procedure does not always converge.

#### B. Choosing Robust Redundant Internal Coordinates

Before describing our alternative procedure for mapping (redundant) internal coordinates to Cartesian coordinates, we should specify the system of internal coordinates that we use. Our method is a refinement of the procedure used in the *Dalton* program.<sup>80</sup> This protocol is, in turn, very similar to the one used in the *Gaussian* program.<sup>70</sup>

#### **1.** Interatomic Distances

Our protocol has five types of interatomic distances: regular (covalent) bonds, hydrogen bonds, inter-fragment bonds, auxiliary bonds, and linear-chain bonds. Regular bonds are included between two atoms,  $\alpha$  and  $\beta$ , if the separation between them is less than 1.3 times the sum of their covalent radii. Next, hydrogen-bonds are assigned by locating all bonds between hydrogen and small electronegative atom that can serve as a hydrogen-bond donor, X = N, O, F, P, S, Cl. For each such bond, one searches to see if there is another electronegative atom, Y = N, O, F, P, S, Cl, that can serve as the hydrogen-bond acceptor. If the H–Y distance is less than 0.9 times the sum of the atoms van der Waals radii *and* the X–H-----Y angle is greater than 90°, then a hydrogen bond is added between this hydrogen atom and Y.

Next, we evaluate whether the system contains multiple fragments. Atoms are assigned to the same fragment if there is a chain of regular bonds between the atoms. If there are multiple fragments in the system, interfragment bonds need to be added between each pair of fragments. If each fragment contains only one atom, then the atom-atom distance is appended to the list of internal coordinates. If one fragment is monatomic but the other fragment is polyatomic, then three coordinates are needed to specify the relative positions of the fragments. If both fragments are polyatomic, then six coordinates are needed to specify their relative positions. In these two cases, at least two interfragment bonds are required. The two shortest interfragment distances are considered to be interfragment bonds. To mitigate the difficulties associated with interfragment bonds disappearing and reappearing during the course of an optimization, additional interfragment bonds are defined for all interfragment distances that are less than 2 Angstroms or 1.3 times the shortest interfragment distance, whichever is larger. To avoid adding too many interfragment bonds, the number of interfragment bonds is not allowed to exceed the number of non-hydrogen atoms in the larger fragment.

In order to represent long linear chains in internal coordinates, one often adds unphysical "ghost atoms" beside the chain so that meaningful dihedral angles can be defined. This is important, for example, when describing the barrier to internal rotations around the molecular axis in polynes like  $Y_3(C-C=C)_n-CX_3$  or the cis-trans isomerization of cumulenes like  $RCH=(C=C)_n=HCR'$ . We have discovered that we can locate the transition states of reactions like these by adding the distance between the first and last atoms of the linear chain. (We tried adding additional inter-chain distances, but this does not seem to be necessary.)

Auxiliary bonds are added when the distance between two atoms is less than 2.5 times the sum of their covalent radii. Most auxiliary bonds correspond to 1,3 interactions. (I.e., these are Urey-Bradley coordinates.<sup>82</sup>) In our approach, auxiliary bonds are never included in bond angles or dihedral angles.

After all of the internal coordinates have been assigned, the rank of the Wilson **B** matrix should be 3N-6 (or 3N-5 for a linear molecule). If the rank of **B** is less than this, the shortest auxiliary bond is upgraded to a regular bond, so that it can be included in angles and dihedrals, and then rank(**B**) is recomputed. This upgrading-procedure for auxiliary bonds is repeated until either the rank of **B** is adequate or all auxiliary bonds are upgraded.

#### 2. Angles between Bonds

The angles between all bonds that are not auxiliary bonds are included as internal coordinates. That is, for each set of three atoms,  $\alpha$ ,  $\beta$ , and  $\gamma$ , in which  $\beta$  is linked to both  $\alpha$  and  $\gamma$  by non-auxiliary bonds,  $\alpha\beta\gamma$  is added to the set of internal coordinates.

#### 3. Dihedral Angles; Rotations Around Bonds

A dihedral angle,  $\alpha\beta\gamma\delta$ , is defined as the angle between the plane defined by the

location of the first three atoms and the plane defined by the location of the last three atoms. A pictorial representation of the dihedral angle is given in Figure 2.1.

**Figure 2.1.** Representation and notational convention for a dihedral angle. Notice that a small change in the Cartesian coordinates of atom  $\alpha$  will induce a large change in the value of the dihedral angle. This demonstrates the problems associated with (nearly) linear dihedrals.



We include dihedral angles for rotations around every non-auxiliary bond. Keeping all possible dihedral angles leads to an explosion in the number of internal coordinates, so we restrict the number of dihedral angles in the following way. Consider a non-auxiliary bond between atoms  $\beta$  and  $\gamma$ . Among all atoms that are bonded to  $\beta$ , let  $\alpha$  be atom that is bonded to the most other atoms. Then the dihedral angles defined by the atoms,  $\alpha\beta\gamma^*$ , where \* denotes any atom bonded to  $\gamma$ , are included in the internal coordinates. Similarly, if atom  $\delta$  is the most-bonded atom that is bonded to  $\gamma$ , all dihedral angles of the form \* $\beta\gamma\delta$  are included in the set of internal coordinates. For example, in an ethane molecule, this approach selects five of the nine possible dihedral angles.

We use improper dihedrals to facilitate the description of puckering motions in planar structures. Consider three atoms,  $\alpha$ ,  $\beta$ , and  $\delta$ , which are all to at most two atoms and which are all bonded to atom  $\gamma$ . The dihedral angle  $\alpha\beta\gamma\delta$  is included if the sum of the angles  $\alpha\gamma\beta$ ,  $\alpha\gamma\delta$ , and  $\beta\gamma\delta$  is greater than 345°.

To define the dihedral angle mathematically, we first define the bond-unit-vectors linking the atoms  $\alpha$ ,  $\beta$ ,  $\gamma$ , and  $\delta$ . E.g., we denote

$$\hat{\mathbf{R}}_{\beta\alpha} = \frac{\mathbf{R}_{\alpha} - \mathbf{R}_{\beta}}{\left|\mathbf{R}_{\alpha} - \mathbf{R}_{\beta}\right|}$$
(2.4)

where  $\mathbf{R}_{\alpha}$  denotes the Cartesian coordinates of atom  $\alpha$ . The plane containing atoms  $\alpha$ ,  $\beta$ , and  $\gamma$  is normal to the vector  $\hat{\mathbf{R}}_{\beta\gamma} \times \hat{\mathbf{R}}_{\beta\alpha}$  and the plane containing atoms  $\beta$ ,  $\gamma$ , and  $\delta$  is normal to the vector  $\hat{\mathbf{R}}_{\gamma\beta} \times \hat{\mathbf{R}}_{\gamma\delta}$ . The dihedral angle between these planes is the same as the angle between the normal-vectors to the planes.

This conventional mathematical representation of dihedral angles fails when atoms  $\alpha$ ,  $\beta$ , and  $\gamma$  (equivalently, atoms  $\beta$ ,  $\gamma$ , and  $\delta$ ) are collinear, because then there is no unique plane containing the atoms. Even when the atoms are not exactly collinear, the conventionally defined dihedral angle is numerically ill-conditioned because a small change in the Cartesian coordinates of one of the atoms will cause a large shift in the direction of the normal vector  $\hat{\mathbf{R}}_{\beta\gamma} \times \hat{\mathbf{R}}_{\beta\alpha}$  and therefore a large change in the dihedral angle. Since a small change in Cartesian coordinate induces a large change in the internal coordinate, the condition number of the Wilson **B** matrix is very large, and the mapping between Cartesian coordinates and internal coordinates becomes numerically illconditioned. This is the problem of linear dihedrals, and it is one of the main reasons why optimizations in internal coordinates sometimes fail unpredictably.

To avoid these instabilities, we have chosen an alternative representation for the dihedral angle. Each dihedral angle is represented by two coordinates, which we call the *robust dihedral descriptors*,

$$\hat{\mathbf{R}}_{\beta\alpha} \cdot \hat{\mathbf{R}}_{\gamma\delta} \tag{2.5}$$

$$\hat{\mathbf{R}}_{\beta\gamma} \cdot \left( \hat{\mathbf{R}}_{\beta\alpha} \times \hat{\mathbf{R}}_{\gamma\delta} \right)$$
(2.6)

Both functions are needed to provide a full description of the dihedral angle. The first function is a replacement for the dihedral cosine; the second is the (signed) volume of the

parallelepiped defined by the bond vectors. When  $\hat{\mathbf{R}}_{\beta\gamma}$  is collinear with one of the other bond vectors, the volume of the parallelepiped is zero. Our tests demonstrate that the robust dihedral descriptors defined by Eqs. (2.5) and (2.6) are not subject to numerical ill-conditioning.

## C. Converting from Redundant Internal Coordinates to Cartesian Coordinates

The robust set internal coordinates presented in the previous section usually provides a highly redundant description of the molecular geometry, with about five times more coordinates than are actually required. A random choice of values for the redundant internal coordinates, then, is very unlikely to represent a feasible molecular geometry. (For example, there is zero probability that a randomly selected choice for the values of the redundant internal coordinates would satisfy the requirement that the sum of the angles in an *n*-atom ring is  $(n-2)\times180^{\circ}$ .) The physically realizable molecular structures,  $\mathbf{q}(\mathbf{x})$ , lie on a 3N-6-dimensional manifold embedded in an  $M_{int}$ -dimensional space. When a point in redundant-internal-coordinates space does not lie on this manifold, there is no corresponding set of Cartesian coordinates.

We wish to map *any* set of redundant internal coordinates to a set of Cartesian coordinates. This mapping will be onto (all sets of Cartesian coordinates are achievable) but not one-to-one, since (a) many choices for the 3N Cartesian coordinates correspond to the same point on the manifold and (b) the dimensionality of the redundant internal space is usually much larger than 3N. The idea is that for the set of redundant internal coordinates of interest,  $\mathbf{q}^{(\text{target})}$ , we find the closest point on the manifold,  $\mathbf{q}(\mathbf{x})$ . We then

assign  $\mathbf{q}^{(\text{target})}$  to a set of Cartesian coordinates that correspond to the closest point on the manifold. Mathematically,

$$\mathbf{x}(\mathbf{q}^{(\text{target})}) = \arg \min_{\mathbf{x}} (\mathbf{q}(\mathbf{x}) - \mathbf{q}^{(\text{target})})^{T} \mathbf{W}(\mathbf{q}(\mathbf{x}) - \mathbf{q}^{(\text{target})})$$
(2.7)

where **W** is a diagonal matrix of weighting factors that is used to adjust the importance of different internal coordinates. We call this approach to assigning Cartesian coordinates to a vector of redundant internal coordinates the *manifold projection method*.

We generally choose **W** to be the identity matrix. However, in constrained optimizations, where certain coordinates are frozen at fixed values, we assign large weights to the frozen coordinates. When we use conventional dihedral angles instead of the robust dihedral descriptors from Eqs. (2.5) and (2.6), we reduce the importance of converging to the target dihedral values when the first-three or last-three atoms are nearly collinear by weighting the dihedral coordinates as  $w_{\angle \alpha\beta\gamma\delta} = \sin^2(\angle \alpha\beta\gamma)\sin^2(\angle \beta\gamma\delta)$ .

To avoid problems with redundancy and near-linear angles (e.g.,  $1^{\circ}$  and  $359^{\circ}$  represent the same bond angle, but the naïvely-computed bond-angle distance is large) we use the cosine of the bond angle to measure the discrepancy of the

$$\left(\cos\left(\theta(\mathbf{x})\right) - \cos\left(\theta^{(\text{target})}\right)\right)^2$$
(2.8)

Similarly, the discrepancy between conventional dihedrals is computed as

$$\left(\cos\left(\phi(\mathbf{x})\right) - \cos\left(\phi^{(\text{target})}\right)\right)^{2} + \left(\sin\left(\phi(\mathbf{x})\right) - \sin\left(\phi^{(\text{target})}\right)\right)^{2}$$
(2.9)

The cost function that is minimized in Eq. (2.7) is therefore

$$\sum_{\text{bonds}} \left( q_{\alpha\beta} \left( \mathbf{x} \right) - q_{\alpha\beta}^{(\text{target})} \right)^{2} + \sum_{\substack{\text{robust}\\\text{dihedrals}}} \left( q_{\alpha\beta\gamma\delta} \left( \mathbf{x} \right) - q_{\alpha\beta\gamma\delta}^{(\text{target})} \right)^{2} \right) \\ + \sum_{\substack{\text{bond}\\\text{angles}}} \left( \left( \cos\left(\theta_{\alpha\beta\gamma}\right) - \cos\left(\theta_{\alpha\beta\gamma}^{(\text{target})}\right) \right)^{2} \right) \\ + \sum_{\substack{\text{conventional}\\\text{dihedrals}}} \sin^{2}\left(\theta_{\alpha\beta\gamma}\right) \sin^{2}\left(\theta_{\beta\gamma\delta}\right) \left( \left( \cos\left(\phi_{\alpha\beta\gamma\delta}\right) - \cos\left(\phi_{\alpha\beta\gamma\delta}^{(\text{target})}\right) \right)^{2} + \left( \sin\left(\phi_{\alpha\beta\gamma\delta}\right) - \sin\left(\phi_{\alpha\beta\gamma\delta}^{(\text{target})}\right) \right)^{2} \right) \right)$$

To generate an initial guess for the optimization, we start with a previously known matched-pair of Cartesian and redundant internal coordinates,  $\mathbf{q}_0$  and  $\mathbf{x}_0$ . The initial guess for the Cartesian coordinates is  $\mathbf{x}^{(guess)} = \mathbf{x}_0 + \mathbf{B}^+ (\mathbf{q}^{(target)} - \mathbf{q}_0)$ . The optimization is solved using Newton's method with a trust radius. We found that it is important to accept steps where the value of the cost function is not changed because the cost function has saddle points, so not every step of the optimization reduces the cost.

#### **D.** An Inexpensive Double-Ended Transition-State Guesser

As a first application for the robust dihedral descriptors and the manifoldprojection method, we have developed three double-ended methods for guessing transition states. The first method is similar to the linear synchronous transit pathway,<sup>83,84</sup> except for the choice of coordinate-system. Specifically, we construct the line-segment between the reactant and the product using redundant internal coordinates

$$(1-p)\mathbf{q}^{(\text{reactant})} + p\mathbf{q}^{(\text{product})}$$
(2.11)

(2.10)

The points on this path generally do not correspond to actual physically realizable molecular geometries, so we use the manifold projection method to construct a pathway from the reactant (p = 0) to the product (p = 1). I.e.,

$$C^{(\text{Method 1})}(p) = \min_{\mathbf{x}} \left| \mathbf{q}(\mathbf{x}) - \left[ (1-p) \mathbf{q}^{(\text{reactant})} + p \mathbf{q}^{(\text{product})} \right] \right|_{\mathbf{w}}^{2}$$
(2.12)

$$\mathbf{x}^{(\text{Method 1})}(p) = \arg \min_{\mathbf{x}} \left| \mathbf{q}(\mathbf{x}) - \left[ (1-p) \mathbf{q}^{(\text{reactant})} + p \mathbf{q}^{(\text{product})} \right] \right|_{\mathbf{w}}^{2}$$
(2.13)

By the triangle inequality,

$$\left| \mathbf{q}(\mathbf{x}) - \left[ (1-p) \mathbf{q}^{(\text{reactant})} + p \mathbf{q}^{(\text{product})} \right] \right|_{\mathbf{w}}^{2}$$
  
$$\leq \left( (1-p) \left| \mathbf{q}(\mathbf{x}) - \mathbf{q}^{(\text{reactant})} \right|_{\mathbf{w}} + p \left| \mathbf{q}(\mathbf{x}) - \mathbf{q}^{(\text{product})} \right|_{\mathbf{w}} \right)^{2} \qquad (2.14)$$

which motivates the choice

$$C^{(\text{Method 2})}(p) = \min_{\mathbf{x}} \left( (1-p) \left| \mathbf{q}(\mathbf{x}) - \mathbf{q}^{(\text{reactant})} \right|_{\mathbf{W}} + p \left| \mathbf{q}(\mathbf{x}) - \mathbf{q}^{(\text{product})} \right|_{\mathbf{W}} \right)^{2}$$
(2.15)

$$\mathbf{x}^{(\text{Method 2})}(p) = \arg\min_{\mathbf{x}} \left( (1-p) \left| \mathbf{q}(\mathbf{x}) - \mathbf{q}^{(\text{reactant})} \right|_{\mathbf{W}} + p \left| \mathbf{q}(\mathbf{x}) - \mathbf{q}^{(\text{product})} \right|_{\mathbf{W}} \right)^2$$

(2.16)

This expression is not differentiable everywhere because magnitudes of vectors appear without being squared. This can be remedied by dropping out the cross-term from Eqs. (2.15) and (2.16). This motivates our final choice,

$$C^{(\text{Method 3})}(p) = \min_{\mathbf{x}} \left( (1-p) \left| \mathbf{q}(\mathbf{x}) - \mathbf{q}^{(\text{reactant})} \right|_{\mathbf{W}}^{2} + p \left| \mathbf{q}(\mathbf{x}) - \mathbf{q}^{(\text{product})} \right|_{\mathbf{W}}^{2} \right)$$

$$(2.17)$$

$$\mathbf{x}^{(\text{Method 3})}(p) = \arg\min_{\mathbf{x}} \left( (1-p) \left| \mathbf{q}(\mathbf{x}) - \mathbf{q}^{(\text{reactant})} \right|_{\mathbf{W}}^{2} + p \left| \mathbf{q}(\mathbf{x}) - \mathbf{q}^{(\text{product})} \right|_{\mathbf{W}}^{2} \right)$$

(2.18)

In all three approaches, we start by constructing the robust redundant internal coordinates of both the reactant and product structures. We then use the union of these reactant-based and product-based coordinates to construct these reaction coordinates in Eqs. (2.13), (2.16), and (2.18).

Conventional synchronous transit methods would then compute the energy at various p values along these reaction coordinates and select the point with the highest energy as an initial approximation to the transition state. The QST2 method would construct an approximate quadratic path, and use a few energy/gradient evaluations along the approximate reaction path to find a better estimate for the transition state.<sup>84</sup> We use a much cheaper, but heuristic, approach. We note that C(p) = 0 for both the reactant and product structures, since, by construction, these sets of internal coordinates correspond to true molecular geometries. It seems plausible, then, that the transition-state structure can be associated with the set of redundant internal coordinates that is furthest from the manifold of realizable molecular geometries. Therefore, we define

$$p_{\max} = \arg\max C(p) \tag{2.19}$$

and use  $\mathbf{x}(p_{\text{max}})$  as an initial guess for the transition state. As shown in appendix 2.1, for Method 2 and Method 3,  $p_{\text{max}} = 0.5$ .

The quantity p represents the fraction of the reaction coordinate that is traversed when one locates the transition state. One can therefore define an expected value for p, a*posteriori*, as

$$p_{\text{post}} = \frac{\left| \mathbf{q}_{\text{TS}} - \mathbf{q}_{\text{product}} \right|}{\left| \mathbf{q}_{\text{TS}} - \mathbf{q}_{\text{reactant}} \right| + \left| \mathbf{q}_{\text{TS}} - \mathbf{q}_{\text{product}} \right|}$$
(2.20)

This value for p can only be computed when the transition-state is already known, but it is useful for assessing the quality of the initial guesses generated by the approximate methods.

## 2.5 Results

#### A. Testing Protocol

We tested the proposed methods for guessing transition states on a database of 130 reactions. The database, which is presented in appendix 2.2, includes a variety of reactions from various sources,<sup>35,85-98,99</sup> including hydrogen-transfer reactions, heavy-atom-transfer reactions, pericyclic reactions, unimolecular reactions, transition metal reactions, etc.. In some cases we functionalized the molecules in the base reaction to construct a more challenging test reaction.

For most of these reactions, the transition state structures were located at the HF/6-31++G(d,p) level. Because Hartree-Fock calculations are notoriously unreliable for molecules including transition metals, reactions 59 and 60 were optimized at the

B3LYP/6-31G(d) level.<sup>90</sup> This choice of basis set and electronic structure methods<sup>100-103</sup> should not be taken as an endorsement of these methods for these reactions, but merely as an indication that these methods are sensible enough to be used for testing generalpurpose transition-state software. Because many of the reactant and product complexes dissociate at the Hartree-Fock level, the reactant and product structures are determined by tracing the intrinsic reaction coordinate from the transition state.<sup>104</sup> Many of the resulting reactant and product structures are multi-fragment molecular complexes, which provides a test for the portion of the redundant internal coordinate algorithm involving multi-fragment molecular systems. Molecular potential energies and gradients were computed with *Gaussian '09*.<sup>105</sup> The methods using internal coordinates and the transition-state-guesser were implemented outside *Gaussian*.

#### **B.** Overview of Results from the Transition-State Guessing Methods

To test the transition-state guessers, the robust redundant internal coordinates were generated for the reactant and product structures using the algorithm presented in section 2.4. The union of these coordinate sets was used to describe the reaction pathway, which was constructed using the three methods from sections 2.5, for p = 0.00, 0.01,0.02,..., 0.99, and 1.00. The value of  $p_{\text{max}}$  (cf. Eq. (2.19)) was computed for all three methods, confirming that only Method 1 ever has  $p_{\text{max}} \neq 0.5$ . The root-mean-square deviation of  $p_{\text{max}}^{(\text{Method 1})}$  from p = 0.5 is 0.018, so even for Method 1, the transition-state guess tends to be near the midpoint of the pathway from reactant to product. The value of  $p_{\text{post}}$  was computed from Eq. (2.20). It is observed that Method 1 has a slight, but significant, tendency to have  $p_{\text{max}}^{(\text{Method }1)} < 0.5$  when  $p_{\text{post}} < 0.5$ , and vice versa. However,  $p_{\text{post}}$  has a much larger range of values, with a root-mean-square deviation from p = 0.5 of 0.095. This confirms the diversity of the reaction database, which contains early, late, and intermediate transition states.

For all three methods, we computed transition-state guesses for both  $p_{max}$  and  $p_{post}$ . We then performed a frequency calculation to ascertain if the transition-state guess was accurate enough for the PES to have one imaginary frequency. As shown in Table 2.1, all three methods locate regions on the PES with just one imaginary frequency for well most of the reactions considered. Methods 1 and 2 perform best, giving predictions in the transition-state region for more than 80% of the reactions. Most failures occur due to multiple imaginary frequencies; this mostly seems to reflect excess bond strain in the transition-state predictions, and does not seem to impair the ability of transition-state algorithms to converge as severely as the absence of any imaginary frequencies (which happens rarely for all three methods).

The root-mean-square deviation of the guessed structures from the true transitionstate structures was computed by aligning the molecules using the Kabsch algorithm.<sup>106,107</sup> First, the structures are translated so that the center of mass of the transition-state guess and the true transition-state structure are both at the origin. Next, one computes the matrix,

$$\mathbf{H} = \sum_{\alpha=1}^{N_{\text{atoms}}} \mathbf{x}_{\alpha}^{(\text{approx})} \left( \mathbf{x}_{\alpha}^{(\text{exact})} \right)^{T} = \sum_{\alpha=1}^{N_{\text{atoms}}} \begin{bmatrix} x_{\alpha}^{(\text{approx})} x_{\alpha}^{(\text{exact})} & x_{\alpha}^{(\text{approx})} y_{\alpha}^{(\text{exact})} & x_{\alpha}^{(\text{approx})} z_{\alpha}^{(\text{exact})} \\ y_{\alpha}^{(\text{approx})} x_{\alpha}^{(\text{exact})} & y_{\alpha}^{(\text{approx})} y_{\alpha}^{(\text{exact})} & y_{\alpha}^{(\text{approx})} z_{\alpha}^{(\text{exact})} \\ z_{\alpha}^{(\text{approx})} x_{\alpha}^{(\text{exact})} & z_{\alpha}^{(\text{approx})} y_{\alpha}^{(\text{exact})} & z_{\alpha}^{(\text{approx})} z_{\alpha}^{(\text{exact})} \end{bmatrix}$$

$$(2.21)$$

and constructs its singular value decomposition,  $\mathbf{H} = \mathbf{U} \Sigma \mathbf{V}^T$ . Assuming that the singular vectors are listed starting with the largest singular value and ending with the smallest singular value, the rotation matrix that maximizes the molecular alignment is

$$\mathbf{R}_{\min} = \mathbf{V} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \det(\mathbf{V}\mathbf{U}^{T}) \end{bmatrix} \mathbf{U}^{T}$$
(2.22)

So the optimal alignment is  $\tilde{\mathbf{x}}_{\alpha}^{(\text{approx})} \leftarrow \mathbf{R}_{\min} \mathbf{x}_{\alpha}^{(\text{approx})}$  and the root-mean-square distance (RMSD) between the structures is then obtained by the definition

rmsd
$$(\tilde{\mathbf{x}}^{(approx)}, \mathbf{x}^{(exact)}) = \sqrt{\frac{1}{N_{atoms}}} \sum_{\alpha=1}^{N_{atoms}} (\tilde{\mathbf{x}}^{(approx)}, -\mathbf{x}^{(exact)}) \cdot (\tilde{\mathbf{x}}^{(approx)}, -\mathbf{x}^{(exact)})$$

(2.23)

The RMSD is quite small (about 0.04 Å) for Methods 1 and 2, and much larger for Method 3. Most transition-state optimization algorithms are very reliable when the guessed transition-state structure is within about 0.05 Å of the true transition state. We tested this by optimizing the geometry with the QST3 algorithm<sup>84</sup> in *Gaussian* '09 using the opt(QST3,calcfc,noeigentest) keywords and the reactant, product, and transition-state

guess structures. On average, less than 15 iterations were required to converge to the transition state.

Figure 2.2 provides a visual representation of the convergence of transition-state optimization. Consistent with the results in Table 2.1, Methods 1 and 2 are the best method for generating transition-state guesses. The performance plot also shows, however, that for "easy cases," where the QST3 algorithm converges in fewer than 10 iterations, Method 1 and Method 3 converge more quickly.

We also considered various ways to further improve the results. First of all, we tried to improve the guess by choosing the weight matrix,  $\mathbf{W}$ , so that the key coordinates (typically the bonds that were being formed/broken in the chemical reaction) were weighted more than the other coordinates in the molecules. (The data we have presented chose  $\mathbf{W}$  to be the identity matrix.) This did not significantly change the results for Method 1 and Method 2, but it dramatically decreased the RMSD for Method 3 by a factor of three (from 0.128 Å to 0.042 Å).

## **Table 2.1.**Assessment of the overall performance of the three different methods of

	<b>Method 1</b> (Eq. (2.13))		<b>Method 2</b> (Eq. (2.16))		Method 3 (Eq. (2.18))	
	$p_{\max}$	$p_{ m post}$	$p_{\text{max}} = 0.5$	$p_{\rm post}$	$p_{\text{max}} = 0.5$	$p_{\mathrm{post}}$
1 imaginary frequency	108	112	109	99	86	112
0 imaginary frequencies	2	2	2	7	3	2
2 or more imaginary frequencies	20	16	19	24	41	16
Mean RMSD in Å	0.0414	0.0408	0.0406	0.0449	0.1281	0.0411
Mean number of steps for TS opt.	9.6	11.2	11.0	12.6	14.5	10.3
# of rxns. converged	118	119	119	119	114	124

guessing transition states.

Second, we wondered if finding a better method for estimating the *p* value would improve the results. Inserting the *a posteriori p* value from Eq. (2.20) provides a reasonable estimate of how much we could improve the results by finding a better model for the choice of *p*. As seen from Table 2.1 and Figure 2.3, choosing  $p_{post}$  vastly improves the accuracy of Method 3, somewhat improves the reliability (in terms of having the appropriate number of imaginary frequencies) of Method 1, and is actually counterproductive for Method 2. Method 3 becomes decisively better for the easier reactions (those which converge in less than 10 steps) and slightly better even for the more difficult cases. With both the approximate ( $p_{max} = .5$ ) and *a posterior* ( $p_{post}$ ) values
of p, more than 90% of the reactions converged within 20 steps using the guess from Method 2.

**Figure 2.2.** A performance plot showing the percentage of reactions (y axis) that have converged within a given number iterations (x axis) for initial transition-state guesses obtained by the three different methods when  $p_{\text{max}}$  (Eq. (2.19)) is chosen.



Figure 2.3. A performance plot showing the percentage of reactions (y axis) that had converged within a given number iterations (x axis) for initial transitionstate guesses obtained by the three different methods when  $p_{post}$ (Eq. (2.20)) is chosen.



## 2.6 Summary

This paper grew out of our attempts to develop a reliable way to transform highly redundant sets of internal coordinates to Cartesian coordinates, for the purpose of transition-state optimization. Conventional methods from the literature failed for some of the reactions we tried (a subset of which are reported in Appendix 2.2), either because of problems with nearly-linear bond angles in dihedrals or because the usual fixed-point iteration method for converting between internal coordinates and Cartesian coordinates failed to converge. We eliminate the former error by introducing robust dihedral descriptors that are not affected by the linear-angle problem, cf. Eqs. (2.5) and (2.6). We avoid the second problem by mapping each possible choice of internal coordinates to the closest point on the (3N-6)-dimensional manifold of allowed internal coordinates, where the manifold is parameterized using the Cartesian coordinates.

This manifold projection technique (cf. Eq. (2.7)) is so robust that even extremely nonphysical sets of redundant internal coordinates can be mapped to Cartesian coordinates. This gave us the courage to consider the internal-coordinate generalization of the linear synchronous transit method (Method 1; Eqs. (2.12)-(2.13)) and, even more daring, the approximate reaction pathways that result when the triangle inequality is applied to the synchronous transit approach (Methods 2 and 3; Eqs. (2.15)-(2.18)). All three methods provide good guesses for the transition state, but Method 1 is particularly promising: more than 80% of the 130 reactions in our database have exactly one imaginary frequency; the root-mean-square distance between the guessed transition-state structure and the true transition state is about 0.04 Å; starting from the guessed structure,

80% of the reactions in our database converge with 20 iterations using the QST3 method. It appears that some improvements could be obtained by finding a protocol for points further from the midpoint of the approximate reaction path for the initial guess. (However, for Methods 1 and 2, it seems likely that subsequent improvements will be small.) All three methods produce reaction paths that pass reasonably near the transition states for the reactions we tested, and are therefore suitable as initial guesses in much more expensive energy-based double-ended methods like string and nudged-elastic-band methods.

## Appendix 2.1: Proof that $p_{max} = 0.5$ for Method 2 and Method 3

In this appendix we demonstrate that  $p_{max} = 0.5$  for Method 2 and Method 3. For simplicity, we will only treat the case where  $\mathbf{W} = 1$ , as that is the case of primary interest to this paper.

The maximum value of p occurs where

$$0 = \frac{d}{dp} \min_{\mathbf{x}} \left( (1-p) \left| \mathbf{q}(\mathbf{x}) - \mathbf{q}^{(\text{reactant})} \right| + p \left| \mathbf{q}(\mathbf{x}) - \mathbf{q}^{(\text{product})} \right| \right)^2$$
(2.24)

Let  $\mathbf{x}(p)$  denote the minimizing value of  $\mathbf{x}$ . Then

$$0 = 2 \Big[ (1-p) |\mathbf{q}(\mathbf{x}(p)) - \mathbf{q}^{(\text{reactant})} | + p |\mathbf{q}(\mathbf{x}(p)) - \mathbf{q}^{(\text{product})} | \Big] \\ \times \Big( - |\mathbf{q}(\mathbf{x}(p)) - \mathbf{q}^{(\text{reactant})} | + |\mathbf{q}(\mathbf{x}(p)) - \mathbf{q}^{(\text{product})} | \Big) \\ + \nabla_{\mathbf{x}} \Big( (1-p) |\mathbf{q}(\mathbf{x}(p)) - \mathbf{q}^{(\text{reactant})} | + p |\mathbf{q}(\mathbf{x}(p)) - \mathbf{q}^{(\text{product})} | \Big)^{2} \cdot \frac{d\mathbf{x}}{dp}$$
(2.25)

The term in square brackets is positive because the reactant and product structures are distinct. The term on the last line is zero because, for the minimizing value of  $\mathbf{x}$  at a specific *p*,

$$0 = \nabla_{\mathbf{x}} \left( (1-p) \left| \mathbf{q} \left( \mathbf{x} \left( p \right) \right) - \mathbf{q}^{(\text{reactant})} \right| + p \left| \mathbf{q} \left( \mathbf{x} \left( p \right) \right) - \mathbf{q}^{(\text{product})} \right| \right)^2$$
(2.26)

Equation (2.25) therefore implies that

$$\left| \mathbf{q}(\mathbf{x}(p)) - \mathbf{q}^{(\text{reactant})} \right| = \left| \mathbf{q}(\mathbf{x}(p)) - \mathbf{q}^{(\text{product})} \right|$$
(2.27)

It is intuitively obvious that if  $\mathbf{q}(\mathbf{x}(p_{\max}))$  is equidistant from the reactant and product structures,  $p_{\max} = 0.5$ . To demonstrate this mathematically, simplify Eq. (2.26) to

$$\mathbf{0} = \left[ (1-p)^{2} + p(1-p) \frac{|\mathbf{q}(\mathbf{x}(p)) - \mathbf{q}^{(\text{product})}|}{|\mathbf{q}(\mathbf{x}(p)) - \mathbf{q}^{(\text{reactant})}|} \right] (\mathbf{q}(\mathbf{x}(p)) - \mathbf{q}^{(\text{reactant})}) \cdot \mathbf{B}$$
$$+ \left[ p^{2} + p(1-p) \frac{|\mathbf{q}(\mathbf{x}(p)) - \mathbf{q}^{(\text{reactant})}|}{|\mathbf{q}(\mathbf{x}(p)) - \mathbf{q}^{(\text{product})}|} \right] (\mathbf{q}(\mathbf{x}(p)) - \mathbf{q}^{(\text{product})}) \cdot \mathbf{B}$$
(2.28)

Substitute the solution (2.27) into Eq. (2.28). One obtains the equation

$$\mathbf{0} = \left[ (1 - p_{\max})^{2} + p_{\max} (1 - p_{\max}) \right] (\mathbf{q} (\mathbf{x} (p_{\max})) - \mathbf{q}^{(\text{reactant})}) \cdot \mathbf{B} + \left[ p_{\max}^{2} + p_{\max} (1 - p_{\max}) \right] (\mathbf{q} (\mathbf{x} (p_{\max})) - \mathbf{q}^{(\text{product})}) \cdot \mathbf{B} \mathbf{0} = (1 - 2p_{\max}) \begin{bmatrix} (1 - p_{\max}) (\mathbf{q} (\mathbf{x} (p_{\max})) - \mathbf{q}^{(\text{reactant})}) \cdot \mathbf{B} \\ - p_{\max} (\mathbf{q} (\mathbf{x} (p_{\max})) - \mathbf{q}^{(\text{product})}) \cdot \mathbf{B} \end{bmatrix}$$
(2.29)

with solution  $p_{\text{max}} = 0.5$ .

The proof that  $p_{\text{max}} = 0.5$  is similar for Method 3. The maximum value of *p* occurs when

$$0 = \frac{d}{dp} \left( \frac{\min_{\mathbf{x}}}{\left( (1-p) \left| \mathbf{q}(\mathbf{x}) - \mathbf{q}^{(\text{reactant})} \right|^2 + p \left| \mathbf{q}(\mathbf{x}) - \mathbf{q}^{(\text{product})} \right|^2 \right) \right)$$
(2.30)

Let  $\mathbf{x}(p)$  denote the minimizing value of  $\mathbf{x}$  from the inner optimization. Then,

$$0 = \frac{d}{dp} \Big( (1-p) \big| \mathbf{q} \big( \mathbf{x}(p) \big) - \mathbf{q}^{(\text{reactant})} \big|^2 + p \big| \mathbf{q} \big( \mathbf{x}(p) \big) - \mathbf{q}^{(\text{product})} \big|^2 \Big)$$
  
=  $- \big| \mathbf{q} \big( \mathbf{x}(p) \big) - \mathbf{q}^{(\text{reactant})} \big|^2 + \big| \mathbf{q} \big( \mathbf{x}(p) \big) - \mathbf{q}^{(\text{product})} \big|^2$   
 $+ \nabla_{\mathbf{x}} \Big( (1-p) \big| \mathbf{q} \big( \mathbf{x}(p) \big) - \mathbf{q}^{(\text{reactant})} \big|^2 + p \big| \mathbf{q} \big( \mathbf{x}(p) \big) - \mathbf{q}^{(\text{product})} \big|^2 \Big) \cdot \frac{d\mathbf{x}(p)}{dp}$ 

(2.31)

The term on the last line is zero for the minimizing value of **x**. This implies that

$$\left|\mathbf{q}\left(\mathbf{x}\left(p_{\max}\right)\right) - \mathbf{q}^{(\operatorname{reactant})}\right|^{2} = \left|\mathbf{q}\left(\mathbf{x}\left(p_{\max}\right)\right) - \mathbf{q}^{(\operatorname{product})}\right|^{2}$$
(2.32)

Since  $\mathbf{q}(\mathbf{x}(p_{\text{max}}))$  is equidistant from the reactant and product structures,  $p_{\text{max}} = 0.5$ .

### **Appendix 2.2: Reaction Database**

This appendix presents the chemical reactions we used for testing the transitionstate guessing approaches presented in this paper. All transition states were determined using HF/6-31++G(d,p) except for reactions 59-60, which were determined using B3LYP/6-31G(d). These reactions are from benchmark reaction sets, primarily those used for validation of density functionals.<sup>85-89,92</sup> For example, reactions 1-28, 63-66, and 71-101 are taken from the database compiled by Goerigk and Grimme.<sup>89</sup> Reactions 29-52 are taken from the collection of Baker and Chan, who propose these reactions as being especially suitable for testing transition-state optimizers.<sup>93</sup> Reaction 53 is taken from our own work on the epoxide hydrolase enzyme.<sup>35,108</sup> Reaction 55 has previously been used by us to test the fast-marching method.<sup>42</sup> Reactions 54,<sup>94</sup> 56-58,<sup>96</sup> and 61<sup>91</sup> come from the meticulous work on potential-energy curves performed by the Toro-Labbe group. Reactions 59 and 60 are from the work of Coyle *et al.*<sup>90</sup> Reaction 62 is from the work of Jenkins.<sup>97</sup> Reactions 102-130 are "decorated" versions of other reactions in the dataset. In these reactions we functionalized molecules in the other reactions. By increasing the conformational space, we hope these reactions will reveal transition-state optimizers that fail to cope with low-barrier conformational changes. Some of these reactions have significant steric constraints.





























Ph.D. Thesis – Sandra Rabi; McMaster University – Chemistry and Chemical Biology







Ph.D. Thesis – Sandra Rabi; McMaster University – Chemistry and Chemical Biology









### 2.7 References

(1) Steinfeld, J. I.; Francisco, J. S.; Hase, W. L.; Prentice Hall Englewood Cliffs, 1989.

(2) Moore, J. W.; Pearson, R. G. 1981, p 455 pp.

(3) Zhou, H. X. *Quarterly Reviews of Biophysics* **2010**, *43*, 219.

(4) E, W. N.; Vanden-Eijnden, E. In Annual Review of Physical Chemistry,

*Vol 61*; Leone, S. R., Cremer, P. S., Groves, J. T., Johnson, M. A., Richmond, G., Eds. 2010; Vol. 61, p 391.

(5) Schlegel, H. B. Wiley Interdisciplinary Reviews-Computational Molecular Science **2011**, *1*, 790.

(6) Farah, K.; Muller-Plathe, F.; Bohm, M. C. *Chemphyschem* **2012**, *13*, 1127.

(7) van Erp, T. S. In *Kinetics and Thermodynamics of Multistep Nucleation and Self-Assembly in Nanoscale Materials: Advances in Chemical Physics, Vol 151*; Nicolis, G., Maes, D., Eds. 2012; Vol. 151, p 27.

(8) Dellago, C.; Bolhuis, P. G. In *Advanced Computer Simulation Approaches* for Soft Matter Sciences Iii; Holm, C., Kremer, K., Eds. 2009; Vol. 221, p 167.

(9) Wales, D. J. *Energy landscapes*; Cambridge University Press: Cambridge,UK ; New York, 2003.

(10) Burkert, U.; Allinger, N. L. J. Comput. Chem. 1982, 3, 40.

(11) Williams, I. H.; Maggiora, G. M. Theochem-Journal of Molecular

Structure 1982, 6, 365.

(12) Scharfenberg, P. Chem. Phys. Lett. 1981, 79, 115.

- (13) Rothman, M. J.; Lohr, L. L. Chem. Phys. Lett. 1980, 70, 405.
- (14) Quapp, W. J. Math. Chem. 2004, 36, 365.
- (15) Hirsch, M.; Quapp, W. J. Comput. Chem. 2002, 23, 887.

(16) Quapp, W.; Hirsch, M.; Imig, O.; Heidrich, D. J. Comput. Chem. **1998**, 19,

1087.

(17) Crehuet, R.; Bofill, J. M.; Anglada, J. M. Theor. Chem. Acc. 2002, 107,

130.

- (18) Bofill, J. M.; Quapp, W. J. Chem. Phys. 2011, 134, 074101.
- (19) Bofill, J. M.; Quapp, W.; Caballero, M. J. Chem. Theory Comp. 2012, 8,

927.

- (20) Liu, Y. L.; Burger, S. K.; Ayers, P. W. J. Math. Chem. 2011, 49, 1915.
- (21) Peters, B.; Heyden, A.; Bell, A. T.; Chakraborty, A. J. Chem. Phys. 2004,

120, 7877.

(22) Quapp, W. Journal of Theoretical & Computational Chemistry 2009, 8,101.

- (23) Quapp, W. J. Chem. Phys. 2005, 122, 174106.
- (24) Cerjan, C. J.; Miller, W. H. J. Chem. Phys. 1981, 75, 2800.
- (25) Banerjee, A.; Adams, N.; Simons, J.; Shepard, R. J. Phys. Chem. 1985, 89,
- 52.

(26) Simons, J.; Jorgensen, P.; Taylor, H.; Ozment, J. J. Phys. Chem. 1983, 87,2745.

(27) Nichols, J.; Taylor, H.; Schmidt, P.; Simons, J. J. Chem. Phys. 1990, 92, 340.

- (28) Simons, J.; Nichols, J. Int. J. Quantum Chem. 1990, 263.
- (29) Baker, J. J. Comput. Chem. 1986, 7, 385.
- (30) Culot, P.; Dive, G.; Nguyen, V. H.; Ghuysen, J. M. Theor. Chim. Act.

**1992**, *82*, 189.

- (31) Besalu, E.; Bofill, J. M. Theor. Chem. Acc. 1998, 100, 265.
- (32) Anglada, J. M.; Bofill, J. M. Int. J. Quantum Chem. 1997, 62, 153.
- (33) Henkelman, G.; Jonsson, H. J. Chem. Phys. **1999**, 111, 7010.
- (34) Heyden, A.; Bell, A. T.; Keil, F. J. J. Chem. Phys. 2005, 123.
- (35) Burger, S. K.; Ayers, P. W. J. Chem. Phys. 2010, 132, 234110.
- (36) Irikura, K. K.; Johnson, R. D. J. Phys. Chem. A 2000, 104, 2191.
- (37) Liu, Y. L.; Burger, S. K.; Dey, B. K.; Sarkar, U.; Janicki, M.; Ayers, P. W.

In Quantum Biochemistry; Matta, C. F., Ed.; Wiley-VCH: Boston, 2010.

- (38) Ohno, K.; Maeda, S. *Phys. Scr.* **2008**, *78*, 058122.
- (39) Maeda, S.; Watanabe, Y.; Ohno, K. Chem. Phys. Lett. 2005, 414, 265.
- (40) Maeda, S.; Ohno, K. J. Phys. Chem. A 2005, 109, 5742.
- (41) Ohno, K.; Maeda, S. Chem. Phys. Lett. 2004, 384, 277.
- (42) Liu, Y.; Ayers, P. W. J. Math. Chem. 2011, 49, 1291.
- (43) Burger, S. K.; Ayers, P. W. J. Chem. Theory Comp. 2010, 6, 1490.
- (44) Burger, S. K.; Liu, Y. L.; Sarkar, U.; Ayers, P. W. J. Chem. Phys. 2009,

130, 024103.

- (45) Dey, B. K.; Janicki, M. R.; Ayers, P. W. J. Chem. Phys. 2004, 121, 6667.
- (46) Dey, B. K.; Ayers, P. W. Mol. Phys. 2006, 104, 541.
- (47) Koslover, E. F.; Wales, D. J. J. Chem. Phys. 2007, 127.
- (48) E, W. N.; Ren, W. Q.; Vanden-Eijnden, E. J. Chem. Phys. 2007, 126,

164103.

- (49) E, W.; Ren, W. Q.; Vanden-Eijnden, E. Phys. Rev. B 2002, 66.
- (50) Ayala, P. Y.; Schlegel, H. B. J. Chem. Phys. 1997, 107, 375.
- (51) Burger, S. K.; Yang, W. J. Chem. Phys. 2007, 127, 164107.
- (52) Burger, S. K.; Yang, W. T. J. Chem. Phys. 2006, 124, 054109.
- (53) Henkelman, G.; Uberuaga, B. P.; Jonsson, H. J. Chem. Phys. 2000, 113,

9901.

- (54) Henkelman, G.; Jonsson, H. J. Chem. Phys. 2000, 113, 9978.
- (55) Trygubenko, S. A.; Wales, D. J. J. Chem. Phys. 2004, 120, 2082.
- (56) Chu, J. W.; Trout, B. L.; Brooks, B. R. J. Chem. Phys. 2003, 119, 12708.
- (57) Maragakis, P.; Andreev, S. A.; Brumer, Y.; Reichman, D. R.; Kaxiras, E.

J. Chem. Phys. 2002, 117, 4651.

- (58) Galvan, I. F.; Field, M. J. J. Comput. Chem. 2008, 29, 139.
- (59) Sheppard, D.; Terrell, R.; Henkelman, G. J. Chem. Phys. 2008, 128,

134106.

- (60) Alfonso, D. R.; Jordan, K. D. J. Comput. Chem. 2003, 24, 990.
- (61) Gonzalez-Garcia, N.; Pu, J. Z.; Gonzalez-Lafont, A.; Lluch, J. M.; Truhlar,

D. G. J. Chem. Theory Comp. 2006, 2, 895.

- (62) Granot, R.; Baer, R. J. Chem. Phys. 2008, 128.
- (63) Ghasemi, S. A.; Goedecker, S. J. Chem. Phys. 2011, 135.
- (64) Fogarasi, G.; Zhou, X. F.; Taylor, P. W.; Pulay, P. J. Am. Chem. Soc.

**1992**, *114*, 8191.

- (65) Pulay, P.; Fogarasi, G. J. Chem. Phys. 1992, 96, 2856.
- (66) Baker, J.; Kessi, A.; Delley, B. J. Chem. Phys. 1996, 105, 192.
- (67) Baker, J.; Pulay, P. J. Chem. Phys. 1996, 105, 11100.
- (68) Baker, J.; Kinghorn, D.; Pulay, P. J. Chem. Phys. 1999, 110, 4986.
- (69) von Arnim, M.; Ahlrichs, R. J. Chem. Phys. **1999**, 111, 9183.
- (70) Peng, C. Y.; Ayala, P. Y.; Schlegel, H. B.; Frisch, M. J. J. Comput. Chem.

**1996**, *17*, 49.

- (71) Baker, J.; Pulay, P. J. Comput. Chem. 2000, 21, 69.
- (72) Farkas, O.; Schlegel, H. B. J. Mol. Struct.: THEOCHEM 2003, 666, 31.
- (73) Farkas, O.; Schlegel, H. B. J. Chem. Phys. 1998, 109, 7100.
- (74) Eckert, F.; Pulay, P.; Werner, H. J. J. Comput. Chem. 1997, 18, 1473.
- (75) Billeter, S. R.; Turner, A. J.; Thiel, W. PCCP 2000, 2, 2177.
- (76) Paizs, B.; Baker, J.; Suhai, S.; Pulay, P. J. Chem. Phys. 2000, 113, 6566.
- (77) Kudin, K. N.; Scuseria, G. E.; Schlegel, H. B. J. Chem. Phys. 2001, 114,

2919.

- (78) Bucko, T.; Hafner, J.; Angyan, J. G. J. Chem. Phys. 2005, 122.
- (79) Doll, K.; Dovesi, R.; Orlando, R. Theor. Chem. Acc. 2006, 115, 354.
- (80) Bakken, V.; Helgaker, T. J. Chem. Phys. 2002, 117, 9160.

(81) Wilson, E. B.; Cross, P. C.; Decius, J. C. *Molecular vibrations : the theory* 

of infrared and Raman vibrational spectra; Dover Publications: New York, 1980.

- (82) Urey, H. C.; Bradley Jr., C. A. *Phys. Rev.* **1931**, *38*, 1969.
- (83) Halgren, T. A.; Lipscomb, W. N. Chem. Phys. Lett. 1977, 49, 225.
- (84) Peng, C. Y.; Schlegel, H. B. Isr. J. Chem. 1993, 33, 449.
- (85) Guner, V.; Khuong, K. S.; Leach, A. G.; Lee, P. S.; Bartberger, M. D.;

Houk, K. N. J. Phys. Chem. A 2003, 107, 11445.

(86) Zhao, Y.; Truhlar, D. G. J. Phys. Chem. A 2004, 108, 6908.

(87) Zhao, Y.; Gonzalez-Garcia, N.; Truhlar, D. G. J. Phys. Chem. A 2005, 109,

2012.

(88) Riley, K. E.; Op't Holt, B. T.; Merz, K. M. J. Chem. Theory Comp. 2007,

3, 407.

(89) Goerigk, L.; Grimme, S. J. Chem. Theory Comp. 2010, 6, 107.

(90) Coyle, J. P.; Johnson, P. A.; DiLabio, G. A.; Barry, S. T.; Muller, J. Inorg.*Chem.* 2010, 49, 2844.

(91) Jaque, P.; Correa, J. V.; De Proft, F.; Toro-Labbe, A.; Geerlings, P.

Canadian Journal of Chemistry-Revue Canadienne De Chimie 2010, 88, 858.

(92) Xu, X. F.; Alecu, I. M.; Truhlar, D. G. J. Chem. Theory Comp. 2011, 7, 1667.

- (93) Baker, J.; Chan, F. R. J. Comput. Chem. 1996, 17, 888.
- (94) Herrera, B.; Toro-Labbe, A. J. Phys. Chem. A 2004, 108, 1830.

(95) Jaque, P.; Toro-Labbe, A.; Politzer, P.; Geerlings, P. *Chem. Phys. Lett.***2008**, *456*, 135.

(96) Guevara-Garcia, A.; Echegaray, E.; Toro-Labbe, A.; Jenkins, S.; Kirk, S.R.; Ayers, P. W. J. Chem. Phys. 2011, 134, 234106.

(97) Mitra, S.; Chandra, A. K.; Gashnga, P. M.; Jenkins, S.; Kirk, S. R. J. Mol.*Model.* 2012, 18, 4225.

(98) Sousa, S. F.; Fernandes, P. A.; Ramos, M. J. J. Phys. Chem. A 2007, 111, 10439.

- (99) S. Rabi, S. K. Burger, and P. W. Ayers, in preparation.
- (100) Becke, A. D. Phys. Rev. A 1988, 38, 3098.
- (101) Becke, A. D. J. Chem. Phys. 1993, 98, 5648.
- (102) Lee, C.; Yang, W.; Parr, R. G. Phys. Rev. B 1988, 37, 785.
- (103) Kohn, W.; Sham, L. J. Phys. Rev. 1965, 140, A1133.
- (104) Fukui, K. Acc. Chem. Res. 1981, 14, 363.
- (105) Frisch, M. J.; Trucks, G. W.; Schlegel, H. B.; Scuseria, G. E.; Robb, M.

A.; Cheeseman, J. R.; Scalmani, G.; Barone, V.; Mennucci, B.; Petersson, G. A.;

Nakatsuji, H.; Caricato, M.; Li, X.; Hratchian, H. P.; Iszmaylov, A. F.; Bloino, J.; Zheng,

G.; Sonnenberg, J. L.; Hada, M.; Ehara, M.; Toyota, K.; Fukuda, R.; Hasegawa, J.; Ishida,

- M.; Nakajima, T.; Honda, Y.; Kitao, O.; Nakai, H.; Vreven, T.; Montgomery Jr., J. A.;
- Peralta, J. E.; Ogliaro, F.; Bearpark, M.; Heyd, J. J.; Brothers, E.; Kudin, K. N.;
- Staroverov, V. N.; Kobayashi, R.; Normand, J.; Raghavachari, K.; Rendell, A.; Burant, J.
- C.; Iyengar, S. S.; Tomasi, J.; Cossi, M.; Rega, N.; Millam, J. M.; Klene, M.; Knox, J. E.;

Cross, J. B.; Bakken, V.; Adamo, C.; Jaramillo, J.; Gomperts, R.; Stratmann, R. E.;

Yazyev, O.; Austin, A. J.; Cammi, R.; Pomelli, C.; Ochterski, J. W.; Martin, R. L.;

Morokuma, K.; Zakrzewski, V. G.; Voth, G. A.; Salvador, P.; Dannenberg, J. J.;

Dapprich, S.; Daniels, A. D.; Farkas, O.; Foresman, J. B.; Ortiz, J. V.; Cioslowski, J.;

Fox, D. J.; Gaussian Inc.: Wallingford CT, 2009.

- (106) Kabsch, W. Acta Crystallogr., Sect. A: Found. Crystallogr. 1976, 32, 922.
- (107) Gower, J. C.; Dijksterhuis, G. B. Procrustes problems; Oxford UP:

Oxford, 2004.

(108) S. Rabi, S. K. Burger, and P. W. Ayers, in preparation.

**Chapter 3** 

# A Reliable Method for Transition-State

# **Optimization Using Partial Computation of the**

# **Hessian and Robust Redundant Internal**

Coordinates

#### 3.1 Overview

We present a novel method for geometry minimization and transition-state optimization using redundant internal coordinates. The salient feature of this method is the identification of certain key chemical coordinates, typically the internal coordinates associated with the bond-breaking/bond-forming processes of interest. The elements of the Hessian associated with these coordinates are approximated with finite differences, while the remainder of the Hessian is updated using quasi-Newton methods. We also utilize a newly proposed description of dihedral angles and an unconventional method for converting from redundant internal coordinates to Cartesian coordinates. We introduce a trust-radius algorithm that uses only gradient information and a method for ensuring that the Hessian matrix has the correct eigenvalue structure. The efficiency of this method for transition-state optimization is assessed using twenty chemical reactions, and results are compared to those obtained using the default *Berny* transition-state optimization method in the *Gaussian* program. Unlike the conventional method, the new approach converges to the targeted transition state in all cases, with comparable computational cost.

## 3.2 Introduction

Computational methods for geometry optimization in quantum chemistry have been under continual development for more than thirty years.<sup>1,2</sup> So why are we developing yet another approach? We are motivated by our frustration with existing methods, which occasionally fail for reasons that are difficult to decipher, much less remedy. Rather than the expensive (in terms of human time and frustration) task of restarting a calculation with a new initial guess or an alternative choice of algorithmadjusting parameters, we would prefer a computational method that converges robustly, even if that method requires slightly more computational effort. The method developed in this chapter does this.

We believe that for geometry minimization, no computational approach should ever fail: any method that systematically makes downhill moves on the molecular potential energy surface (PES) should eventually converge to the nearest local minimum. For geometry minimization, then, the only acceptable failure is a failure of the quantum chemistry method used to evaluate the gradient. (Actually, we also find these failures unacceptable, but fixing these problems is beyond the scope of this work.) Robust methods for redundant internal coordinates, recently introduced by us, move us closer to this goal.<sup>3</sup>

Transition-state optimization is much more difficult than geometry minimization because a preferred move could increase or decrease the energy and, if the initial guess for the transition state is poor, the norm of the gradient may need to increase during the optimization. Unfortunately, it is much easier to generate good initial geometries for stable structures (geometry minimization) than it is for transition states. Transition state structures are sometimes counterintuitive, and are rarely predictable using simple force-field-type approaches.<sup>4</sup> This has motivated an entirely different approach to transition-state optimization, where entire reaction paths<sup>5-21</sup> (or at least the reactant and product<sup>22,23,24</sup>) or even entire regions of the PES are computed<sup>25-35</sup> in pursuit of the transition state. Such methods are more robust, but their cost grows exponentially with
the dimensionality of the problem, and often require many thousands (or more) evaluations of the PES. In this work, we are primarily focussed on the conventional transition-state-finding approaches.

Failures in transition-state optimization usually occur because (1) the underlying quantum chemistry method fails to compute the gradient, (2) the system of internal coordinates, or a move computed in internal coordinates, is unphysical, (3) the (approximate) Hessian matrix is inaccurate, leading to poor optimization steps, (4) the method converges, but converges to a transition state other than the one of interest (e.g., one corresponding to a conformational change, instead of bond-breaking) or to a stationary point on the PES with more (or less) than one imaginary frequency. The first problem is more severe for transition states because failures in self-consistent-field methods are most probable when the HOMO-LUMO gap is small, and this is more probable near transition states, which often correspond to avoided crossings between ground- and excited-state potential energy surfaces. This problem is beyond the scope of this work. The second problem can be solved using the robust redundant internal coordinate methods we have recently presented.<sup>36</sup> In this work, we address the third and fourth problems by having the user specify certain key chemical coordinates associated with the reaction of interest; we then evaluate the elements of the Hessian that are associated with these coordinates accurately (using finite differences). A reliable approach for automatically selecting the chemical coordinates will be presented in chapter 6.

118

The basic idea of using finite-difference information for coordinates associated with the chemical reaction is not new. We have previously implemented such an approach using the *z*-matrix or Cartesian coordinates of reactive atoms;<sup>37</sup> we use redundant internal coordinates in this work. The QST2 and QST3 methods of Peng and Schlegel are similar,<sup>23</sup> as are various approaches in the family of dimer methods.<sup>38,39</sup> (Both these methods selectively refine the portion of the quasi-Newton Hessian associated with the reaction coordinate.)

While the focus of our research is on developing robust transition state optimization approaches, we wish to do so with minimal loss of efficiency. The efficiency of transition-state optimization methods is controlled by several factors, including the choice of internal coordinates (we use redundant internal coordinates), the choice of initial Hessian (we use the exact Hessian to initialize the algorithm), the choice of quasi-Newton Hessian updating method, and the method of step-size control. We explore several different Hessian-updating approaches in this work, all of which use finite differences in a reduced-dimensional space associated with the key chemical coordinates governing the reaction of interest. We explore three trust-radius-based approaches (including the popular rational function optimization method) for controlling the stepsize.

In section 3.3, we present the details of our new approach to transition-state optimization, as implemented in a program we call *Saddle*. Section 3.4 is a brief overview of *Saddle*, leading into section 3.5, where different variants of *Saddle* are tested against the standard Berny transition-state optimizer,<sup>40</sup> as implemented in the *Gaussian* 

program.<sup>41</sup> In this chapter, we consider a set of twenty reactions, including some of the frequently tested (and quite challenging) reactions proposed by Baker.<sup>42</sup> We find that our best methods are more robust (they converge more frequently) than the *Berny* optimizer, but they require a few additional gradient evaluations. Chapter 4 considers a systematic test for many different initial guesses (of various quality) and a diverse test-set of 131 reactions. Unless otherwise noted, all quantities in this chapter are reported in atomic units.

# 3.3 Key Components of the *Saddle* Algorithm

#### A. Initialization

The efficiency and robustness of transition-state optimizers is sensitive to the initial guess for the transition-state structure and the initial choice for the Hessian matrix. Finding a good initial guess is particularly difficult for large and floppy molecules, because there are many low-frequency motions and low-barrier pathways on the PES associated with conformational changes, rather than chemical reactions. We recently developed a very efficient approach for constructing initial guesses for transition-state optimization based on interpolating between the reactant and product structures in redundant internal coordinates.<sup>43</sup> This approach gives good estimates for the reactions considered here.

Our method requires the user to provide an initial guess for the molecular structure of the transition state in Cartesian coordinates, along with an initial guess for the Hessian in Cartesian coordinates. In the tests reported here, we always initialize the method by calculating the exact Hessian for the initial structure. We have observed, however, that the method works equally well when initializing with Hartree-Fock calculations in a minimal basis set, and that it converges, albeit more slowly, when the initial Hessian is taken from a semiempirical method. We will present our results for initial Hessian approximations in chapter 5.

# B. Choice of robust redundant internal coordinates

The efficiency of geometry optimization methods is sensitive to the choice of coordinate-system for the molecular potential energy surface.<sup>44,45</sup> The most straightforward system of coordinates, and the one to which most computational chemists revert when all else fails, is the system of Cartesian coordinates constructed from the  $X_{\alpha}$ ,  $Y_{\alpha}$ , and  $Z_{\alpha}$  coordinates for each atom  $\alpha$ . Geometry optimization in Cartesian coordinates is relatively inefficient, and so more natural systems based on internal coordinates, including internuclear (bond) distances, bond angles, and dihedral rotations around bonds, are often used. The Z-matrix is the conventional system of internal coordinates.<sup>46</sup> The problem with the Z-matrix approach is that there are many different Z-matrices for the same molecular structure, depending on the order in which the atoms are listed, and the performance of geometry optimization methods deteriorates if a poor atom-ordering is chosen. For molecules containing rings, there is no way to treat each atom in the ring equivalently using a Z-matrix; Z-matrices work so poorly in this case that Cartesian coordinates are usually better.<sup>44</sup>

Following the work of Pulay, in this work we use a system of redundant internal coordinates.<sup>47</sup> In this approach, we overdetermine the molecular geometry by specifying even more bond distance, angles, and dihedral torsions than are required to fully specify the molecular geometry. Overall, redundant internal coordinates is the most efficient coordinate system for geometry optimization.<sup>48-63</sup>

The redundancy of coordinates can either be dealt with directly or the redundancy can be eliminated by selecting a linearly independent set of 3N - 6 (3N - 5 for a linear molecule) coordinates; these coordinates are generally linear combinations of the fundamental, chemically intuitive, redundant internal coordinates. In our work, we have tried both approaches, and the algorithm we present does not seem to be especially sensitive to the approach we take. Our simplest method, which is presented in this chapter, is based on a system of coordinates that is very closely related to the delocalized internal coordinates that Baker proposed,<sup>50</sup> partly as a computationally and algorithmically feasible approach to Pulay's natural internal coordinate set.<sup>47</sup>

The system of robust redundant internal coordinates we use is fully described in chapter 2. Here we mention only its salient features. It includes (covalent) bond lengths, hydrogen-bond lengths, internuclear distances between fragments, and end-to-end internuclear distances for linear chains. Bond-angle and dihedral-angle coordinates are built using these types of bonds. The specific criteria for defining these coordinates is based on the system of redundant internal coordinates developed by Peng and Schlegel for the *Gaussian* program,<sup>55</sup> and augmented by auxiliary bonds (to describe 1,3 interactions), as proposed by Bakken and Helgaker for the *Dalton* program.<sup>4</sup> Auxiliary

bonds are usually not included in bond-angles or dihedral-angles, but we include anglecoordinates using auxiliary bonds when this is needed to ensure a complete set of 3N - 6(or 3N - 5) coordinates. Our minor revisions to Bakken and Helgaker's protocol include an algorithm for assigning interfragment bonds that ensures that the relative orientation of the fragments is always fully specified and the decision to add end-to-end distances for molecules containing long linear chains.

All interatomic lengths are measured in atomic units (1 Bohr = 0.52917725 Angstroms). Rather than using bond angles directly as internal coordinates, we use the cosine of the bond angles. This improves the robustness of the program when  $\alpha\beta\gamma$  is nearly linear.

The most important change in our robust redundant coordinate system is the way we define dihedral angles. For each non-auxiliary bond,  $\beta\gamma$ , the atom  $\alpha$  that is bonded (not with an auxiliary bond) to  $\beta$  and bonded to the most other atoms besides  $\beta$  is identified. Then all possible dihedrals angles,  $\alpha\beta\gamma^*$ , where \* is a wild card referring to any atom that is bonded to  $\gamma$  by a non-auxiliary bond are added to the system of internal coordinates. Similarly, the dihedrals defined by \* $\beta\gamma\delta$  are included, where  $\delta$  is the most-connected atom that is bonded to  $\gamma$ . This set of dihedral angles tends to be larger than the set used by Bakken and Helgaker, but except for very large molecules, this does not present a problem.

The problem of linear dihedrals refers to the fact that when either the bond angle  $\alpha\beta\gamma$  or the bond angle  $\beta\gamma\delta$  is nearly linear (e.g.,  $\sin(\alpha\beta\gamma) \approx 0$ ), then the dihedral angle is

123

ill-defined since a small change in the Cartesian coordinates can cause an enormous change in the dihedral angle. To avoid numerical difficulties associated with linear dihedrals, instead of describing torsions with dihedral angles, we use the alternative dihedral coordinates,

$$\hat{\mathbf{R}}_{\beta\alpha} \cdot \hat{\mathbf{R}}_{\gamma\delta} \tag{3.1}$$

$$\hat{\mathbf{R}}_{\beta\gamma} \cdot \left( \hat{\mathbf{R}}_{\beta\alpha} \times \hat{\mathbf{R}}_{\gamma\delta} \right) \tag{3.2}$$

where

$$\hat{\mathbf{R}}_{\alpha\beta} = \frac{\mathbf{R}_{\beta} - \mathbf{R}_{\alpha}}{\left|\mathbf{R}_{\beta} - \mathbf{R}_{\alpha}\right|}$$
(3.3)

is the unit vector connecting atoms  $\alpha$  and  $\beta$ . Using the robust dihedral descriptors from Eqs. (3.1) and (3.2) prevents many of the problems other approaches have when converting internal coordinates to Cartesian coordinates. In all of our testing, we have never observed a problem with coordinate transformation between robust redundant internal coordinates and Cartesian coordinates.

# C. Coordinate transformations between Cartesian and redundant internal coordinates

The Wilson **B** matrix is the Jacobian matrix for the transformation from the  $3N_{\text{atoms}}$  Cartesian coordinates, denoted  $\{x_j\}_{j=1}^{3N_{\text{atoms}}}$ , to the  $N_{\text{int}}$  internal coordinates, denoted  $\{q_i\}_{i=1}^{N_{\text{int}}}$ .

Computing the elements of the Jacobian matrix,

$$b_{ij} = \frac{\partial q_i}{\partial x_i} \tag{3.4}$$

is a straightforward but tedious exercise in trigonometry. The **B** matrix is rectangular (usually there are far more rows than columns) and singular because the molecule's geometry is specified by just  $3N_{\text{atoms}} - 6$  ( $3N_{\text{atoms}} - 5$  for a linear molecule) internal coordinates. The nonsingular vectors of **B** are called the delocalized internal coordinates.<sup>50,52</sup>

By definition, the Wilson **B** matrix allows one to express a small change in Cartesian coordinates as a small change in internal coordinates,

$$\mathbf{B}\boldsymbol{\delta}\mathbf{x} = \boldsymbol{\delta}\mathbf{q} \tag{3.5}$$

The inverse of this transformation is nonunique because the redundant internal coordinates do not specify the location or orientation of the molecule, while the Cartesian coordinates do. We decided that a change in internal coordinates should preserve the position and orientation of the molecule and, therefore, that the smallest Cartesian move that is consistent with a given perturbation of the internal coordinates should be preferred. This suggests that we should prefer the inverse transformation with minimum norm, which is the Moore-Penrose pseudoinverse of  $\mathbf{B}$ ,

$$\mathbf{B}^+ \boldsymbol{\delta} \mathbf{q} = \boldsymbol{\delta} \mathbf{x} \tag{3.6}$$

Not every change in redundant internal coordinates is physically allowable. For example, in a three atom ring, most changes in internal coordinates will violate the constraint that  $\alpha\beta\gamma + \beta\alpha\gamma + \beta\gamma\alpha \neq 180^\circ$ . We can project an unrealizable change of

redundant internal coordinates into its realizable component by first converting the change to Cartesian coordinates, and then converting it back. I.e.,

$$\delta \tilde{\mathbf{q}} = \mathbf{P} \delta \mathbf{q} \tag{3.7}$$

with

$$\mathbf{P} = \mathbf{B}\mathbf{B}^+ \tag{3.8}$$

In numerical optimization, we also need expressions for the derivatives of the molecular potential energy with respect to Cartesian and internal coordinates. We denote the energy gradients in Cartesian and internal coordinates as  $\mathbf{g}_x$  and  $\mathbf{g}_q$ , respectively. Similarly, we denote the second-derivative matrices (Hessians) in Cartesian and internal coordinates as  $\mathbf{H}_x$  and  $\mathbf{H}_q$ , respectively. Then, from the definition of  $\mathbf{B}$ ,

$$\mathbf{g}_{x} = \mathbf{B}^{T} \mathbf{g}_{q} \tag{3.9}$$

$$\mathbf{H}_{x} = \mathbf{B}^{T} \mathbf{H}_{q} \mathbf{B} + \mathbf{K}$$
(3.10)

where **K** is given by,

$$K_{jk} = \sum_{i} \left[ \mathbf{g}_{q} \right]_{i} b'_{ijk}$$
(3.11)

where  $[\mathbf{g}_q]_i$  denotes the *i*<sup>th</sup> component of the gradient in internal coordinates and

$$b'_{ijk} \equiv \frac{\partial^2 q_i}{\partial x_i \partial x_k} = \frac{\partial b_{ij}}{\partial x_k}$$
(3.12)

denotes the derivative of an element of the Wilson **B** matrix. Applying  $\mathbf{B}^+$  to equations (3.9) and (3.10) gives equations for energy derivatives in internal coordinates,

$$\mathbf{g}_q = \left(\mathbf{B}^T\right)^+ \mathbf{g}_x \tag{3.13}$$

$$\mathbf{H}_{q} = \left(\mathbf{B}^{T}\right)^{+} \left(\mathbf{H}_{x} - \mathbf{K}\right)\mathbf{B}^{+}$$
(3.14)

These expressions are required because quantum chemistry packages compute derivatives in Cartesian coordinates. All of the preceding equations are standard,<sup>4,49</sup> and are included here only for the sake of completeness.

Because the mapping between Cartesian and internal coordinates is nonlinear, Eqs. (3.5) and (3.6) are only valid for infinitesimal geometric changes. Converting a point in Cartesian coordinates to the corresponding point in the space of redundant internal coordinates,  $\mathbf{q}(\mathbf{x})$ , is a straightforward exercise in trigonometry. The inverse transformation is not always defined, because there is only a  $(3N_{\text{atoms}} - 6)$ -dimensional manifold of realizable redundant internal coordinates. If one starts with a point on this manifold,  $\mathbf{q}_k$ , and takes the noninfinitesimal step mandated by some optimization procedure,

$$\mathbf{q}_{k+1} = \mathbf{q}_k + \mathbf{s}_k \tag{3.15}$$

the resulting point will rarely be on the manifold. That is, the set of Cartesian coordinates computed using Eq. (3.6),

$$\mathbf{x} = \mathbf{x}_k + \mathbf{B}^* \mathbf{s}_k \tag{3.16}$$

will usually not exactly correspond to the target set of internal coordinates. We choose  $\mathbf{x}_{k+1}$  by finding the point,  $\mathbf{q}(\mathbf{x})$ , on the manifold of realizable internal coordinates that is closest to the target value,  $\mathbf{q}_{k+1}$  from Eq. (3.15). To do this, we use Eq. (3.16) as an initial guess for  $\mathbf{x}_{k+1}$  and then minimize the distance in redundant internal space,

$$\mathbf{x}_{k+1} \equiv \arg\min_{\mathbf{x}} |\mathbf{q}(\mathbf{x}) - \mathbf{q}_{k+1}|^2.$$
(3.17)

Details of this procedure can be found in chapter 2. Equation (3.17) is more reliable than the conventional iterative procedures for converting from redundant internal to Cartesian coordinates.<sup>4,47,49,50,54,55,64</sup>

#### D. On-the-fly addition of new internal coordinates

Occasionally a change in molecular geometry during the optimization process causes the set of robust redundant internal coordinates to change (cf. 3.3.B and chapter 2). We never remove internal coordinates during the optimization, though we have observed that having excess redundant internal coordinates can cause performance to deteriorate slightly. However, when our coordinate-selection procedure suggests that a new internal coordinate should be added, we add it.

When new coordinates are added, we need to compute the Hessian in the new set of internal coordinates. To do this, we use

$$\mathbf{H}_{q}^{(\text{new})} = \left(\mathbf{B}^{(\text{new})}\right)^{+} \mathbf{B}^{(\text{old})} \mathbf{H}_{q}^{(\text{old})} \left(\left(\mathbf{B}^{(\text{new})}\right)^{+} \mathbf{B}^{(\text{old})}\right)^{T}$$
(3.18)

This is equivalent to transforming the Hessian from the old set of internal coordinates to Cartesian coordinates, and then transforming the Cartesian Hessian to the new set of redundant internal coordinates. In performing this transformation, there is the possibility of discarding useful information that has been built up from the quasi-Newton updates. This motivated our decision to allow important internal coordinates to be added during the course of the optimization, but never to allow coordinates (which might then need to be added, again, in a later iteration) to be removed.

#### E. "Reduced" coordinates to describe key chemical changes

Most chemical reactions can be characterized by a few key internal coordinates, typically those associated with bond-breaking and bond-forming processes, as well as conformational changes that are tightly coupled to the reaction coordinate. This has motivated us, and others, to use reduced-dimensionality potential energy surfaces (in which the potential energy is minimized with respect to all of the "spectator coordinates") to characterize chemical reactions.<sup>32,65-68</sup> We adopt the same perspective here: the user is expected to identify key internal coordinates that are involved in the elementary chemical reaction whose transition-state is being targeted. Our goal is to ensure that the PES is accurately described for the coordinates. We will achieve this by using finite differences to approximate the elements of the Hessian matrix associated with these key coordinates.

If the user does not wish to manually choose internal coordinates, she can provide the molecular structures of the reactant and product. A set of internal coordinates is generated by the union of the sets of internal coordinates for the reactant, product, and transition-state-guess structures. An interatomic distance is selected as a reduced coordinate if its length changes by more than half the sum of the covalent radii of its composing atoms when comparing any two of the three available structures (reactant/product, reactant/transition-state guess, or product/transition-state guess). An angle,  $\alpha\beta\gamma$ , is treated as a reduced coordinate if (a) atoms  $\alpha$ ,  $\beta$ , and  $\gamma$  are all involved in at least one reduced bond-length coordinate and (b) the angle changes by at least 30° between the reactant, product, and transition-state structures. We could not find a simple and reliable criterion for selecting dihedral angles, so these must be manually selected. We will report results on the automatic generation of reduced coordinates separately. For the reactions considered in this chapter, all the reduced coordinates were selected manually. (The reactions considered, and their associated reduced coordinates, are presented in Appendix 3.2.) After selecting the R reduced coordinates, we reorder the list of redundant internal coordinates so that they appear first.

# F. The V matrix of delocalized reduced and internal coordinates

The geometry optimization method we have developed uses a set of  $3N_{\text{atoms}} - 6$  $(3N_{\text{atoms}} - 5 \text{ for a linear molecule})$  nonredundant delocalized orthogonal internal coordinates; this is analogous to the set proposed by Baker,<sup>50,52</sup> but is different because the reduced coordinates are treated separately. The nonredundant coordinates, denoted  $\left\{v^{(j)}\right\}_{j=1}^{3N_{\text{atoms}}-6(5)}$ , are linear combinations of the redundant internal coordinates,

$$v^{(j)} = \sum_{i=1}^{N_{\text{int}}} v_i^{(j)} q_i$$
(3.19)

The Jacobian for the transformation from delocalized internal coordinates to redundant internal coordinates is a  $N_{\text{int}} \times 3N_{\text{atoms}} - 6(5)$  dimensional matrix, denoted

$$\mathbf{V} = \begin{bmatrix} v_1^{(1)} & v_1^{(2)} & \cdots & v_1^{(3N_{\text{atoms}} - 6(5))} \\ v_2^{(1)} & v_2^{(2)} & \cdots & v_2^{(3N_{\text{atoms}} - 6(5))} \\ \vdots & \vdots & & \vdots \\ v_{N_{\text{int}}}^{(1)} & v_{N_{\text{int}}}^{(2)} & \cdots & v_{N_{\text{int}}}^{(3N_{\text{atoms}} - 6(5))} \end{bmatrix}$$
(3.20)

The first step in the construction of the V matrix is the generation of delocalized internal coordinates. We choose the  $3N_{\text{atoms}} - 6(5)$  non-singular vectors from the singular value decomposition of the Wilson **B** matrix, which we denote

$$\mathbf{a}^{(i)} = \begin{bmatrix} a_1^{(i)}, a_2^{(i)}, \dots, a_{N_{\text{int}}}^{(i)} \end{bmatrix}^T \qquad i = 1, 2, \dots, 3N_{\text{atoms}} - 6(5) \qquad (3.21)$$

Though our procedure for generating these coordinates is different, these vectors span the same space as Baker's delocalized internal coordinates because they are eigenvectors of  $\mathbf{BB}^{T\,50,52}$ .

Making a small change in one of the reduced redundant internal coordinates, without changing any of the other internal coordinates, usually gives an unrealizable set of redundant internal coordinates. Therefore, we first consider only realizable perturbations of the reduced coordinates (cf. Eqs. (3.7) and (3.8))

$$\mathbf{b}^{(j)} = \mathbf{P}\hat{\mathbf{e}}^{(j)} = \mathbf{B}\mathbf{B}^{+}\hat{\mathbf{e}}^{(j)}$$
  $j = 1, 2, \dots R$  (3.22)

Here  $\hat{\mathbf{e}}^{(j)}$  is a unit vector with elements  $\delta_{ij}$ ; it is zero except for a 1 in the  $j^{\text{th}}$  position. (Remember that we chose to order the internal coordinates with the reduced coordinates appearing first.)

The vectors  $\mathbf{b}^{(j)}$  are not orthogonal. We orthogonalize them by forming the Grammian,

$$\mathbf{G} = \begin{bmatrix} \mathbf{b}^{(1)} \cdot \mathbf{b}^{(1)} & \mathbf{b}^{(1)} \cdot \mathbf{b}^{(2)} & \cdots & \mathbf{b}^{(1)} \cdot \mathbf{b}^{(R)} \\ \mathbf{b}^{(2)} \cdot \mathbf{b}^{(1)} & \mathbf{b}^{(2)} \cdot \mathbf{b}^{(2)} & \cdots & \mathbf{b}^{(2)} \cdot \mathbf{b}^{(R)} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{b}^{(R)} \cdot \mathbf{b}^{(1)} & \mathbf{b}^{(R)} \cdot \mathbf{b}^{(2)} & \cdots & \mathbf{b}^{(R)} \cdot \mathbf{b}^{(R)} \end{bmatrix}$$
(3.23)

An orthogonal basis for the reduced space can be constructed from the Grammian's eigenvectors,  $\mathbf{Gg}^{(j)} = \lambda^{(j)}\mathbf{g}^{(j)}$ . Specifically the delocalized internal coordinates corresponding to the reduced space are defined as

$$\tilde{\mathbf{v}}^{(j)} = \sum_{r=1}^{R} g_r^{(j)} \mathbf{b}^{(r)} \qquad j = 1, 2, \dots, R \qquad (3.24)$$

If the Grammian has zero (or negligibly small) eigenvalues, this indicates that the reduced space has redundancies. In this case, we reduce the dimensionality of the reduced space by including only the  $\mathbf{g}^{(j)}$  with nonzero eigenvalues.

The projection operator onto the reduced space is:

$$\mathbf{P}_{\text{reduced}} \mathbf{a}^{(j)} = \sum_{r=1}^{R} \left( \mathbf{a}^{(j)} \cdot \tilde{\mathbf{v}}^{(r)} \right) \tilde{\mathbf{v}}^{(r)}$$
(3.25)

and the projection onto the non-reduced space is therefore

$$\mathbf{d}^{(j)} \equiv \left(\mathbf{I} - \mathbf{P}_{\text{reduced}}\right) \mathbf{a}^{(j)} = \mathbf{a}^{(j)} - \sum_{r=1}^{R} \left(\mathbf{a}^{(j)} \cdot \mathbf{v}^{(r)}\right) \mathbf{v}^{(r)}$$
(3.26)

The vectors  $\mathbf{d}^{(j)}$  are not orthogonal. We orthogonalize them by forming the Grammian,

$$\mathbf{F} = \begin{bmatrix} \mathbf{d}^{(1)} \cdot \mathbf{d}^{(1)} & \mathbf{d}^{(1)} \cdot \mathbf{d}^{(2)} & \cdots & \mathbf{d}^{(1)} \cdot \mathbf{d}^{(3N_{atoms} - 6(5))} \\ \mathbf{d}^{(2)} \cdot \mathbf{d}^{(1)} & \mathbf{d}^{(2)} \cdot \mathbf{d}^{(2)} & \cdots & \mathbf{d}^{(2)} \cdot \mathbf{d}^{(3N_{atoms} - 6(5))} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{d}^{(3N_{atoms} - 6(5))} \cdot \mathbf{d}^{(1)} & \mathbf{d}^{(3N_{atoms} - 6(5))} \cdot \mathbf{d}^{(2)} & \cdots & \mathbf{d}^{(3N_{atoms} - 6(5))} \cdot \mathbf{d}^{(3N_{atoms} - 6(5))} \end{bmatrix}$$
(3.27)

The Grammian has *R* zero eigenvalues. Using only the eigenvectors,  $\mathbf{F}\mathbf{f}^{(j)} = \lambda^{(j)}\mathbf{f}^{(j)}$ , of the Grammian with nonzero eigenvalues, we can construct an orthogonal basis for the nonreduced space,

$$\tilde{\mathbf{v}}^{(R+j)} = \sum_{i=1}^{3N_{\text{atoms}}-6(5)} \mathbf{f}_{i}^{(j)} \mathbf{d}^{(i)} \qquad \qquad j = 1, 2, \dots 3N_{\text{atoms}} - 6(5) - R; \lambda^{(j)} > 0$$
(3.28)

At each iteration in the optimization process, the  $\tilde{\mathbf{V}}$  matrix constructed from the vectors in Eqs. (3.24) and (3.28) is constructed using the preceding procedure, Eqs. (3.20) -(3.28). The specific choice of basis for the nonredundant delocalized orthogonal coordinates is almost arbitrary, so after we have determined  $\tilde{\mathbf{V}}$ , we allow rotation of the basis so that the alignment with the previous  $\mathbf{V}$  matrix is maximized. That is, we find the orthogonal matrix,  $\mathbf{Q}$ , for which

$$\mathbf{Q}^{(\min)} \equiv \arg \min_{\left\{\mathbf{Q} \mid \mathbf{Q}\mathbf{Q}^{T} = \mathbf{Q}^{T}\mathbf{Q} = \mathbf{I}\right\}} \left(\tilde{\mathbf{V}}\mathbf{Q} - \mathbf{V}^{(\text{old})}\right)^{T} \left(\tilde{\mathbf{V}}\mathbf{Q} - \mathbf{V}^{(\text{old})}\right)$$
(3.29)

This is a Procrustes problem. It is solved by maximizing the overlap between the basis sets. Using the singular-value decomposition of the overlap matrix

$$\mathbf{S} = \tilde{\mathbf{V}}^T \mathbf{V}^{(\text{old})} = \mathbf{U} \boldsymbol{\Sigma} \mathbf{W}^T, \qquad (3.30)$$

the optimal orthogonal matrix is

$$\mathbf{Q}^{(\min)} = \mathbf{U}\mathbf{W}^T \tag{3.31}$$

and the maximally aligned basis is

$$\mathbf{V} = \tilde{\mathbf{V}}\mathbf{Q} \,. \tag{3.32}$$

This procedure can be criticized because it allows the reduced coordinates and the nonreduced coordinates to mix. We did not observe this to be problematic.

At each iteration, we map the molecular structure in redundant internal coordinates,  $\mathbf{q}$ , onto a set of nonredundant delocalized internal coordinates,  $\mathbf{v}$ , that are linear combinations of the redundant internals. That is, for this iteration, we consider  $\mathbf{V}$  to be fixed, rather than as a function of  $\mathbf{q}$ . This makes the conversion of gradients, Hessians, and steps from redundant internals to nonredundant internals straightforward. Specifically, we have

$$\mathbf{g}_{v} = \mathbf{V}^{T} \mathbf{g}_{q} \tag{3.33}$$

$$\mathbf{g}_q = \mathbf{V}\mathbf{g}_v \tag{3.34}$$

$$\mathbf{H}_{v} = \mathbf{V}^{T} \mathbf{H}_{q} \mathbf{V}$$
(3.35)

$$\mathbf{H}_{q} = \mathbf{V}\mathbf{H}_{\nu}\mathbf{V}^{T} \tag{3.36}$$

$$\Delta \mathbf{v} = \mathbf{V}^T \Delta \mathbf{q} \tag{3.37}$$

$$\Delta \mathbf{q} = \mathbf{V} \Delta \mathbf{v} \tag{3.38}$$

If we considered **V** to be a function of **q** within each iteration, then transforming a step from **v**-coordinates to **q**-coordinates would require a similar procedure to Eq. (3.17). More problematic, we would need to construct a matrix analogous to the **K** matrix in Eq. (3.14), requiring us to compute

$$\frac{\partial \mathbf{V}^{T}}{\partial q_{k}} = \frac{\partial^{2} v^{(i)}}{\partial q_{i} \partial q_{k}}.$$
(3.39)

This could be computed, but we view it as an undesirable complication. Therefore we merely treat  $\mathbf{V}$  as a computational tool for specifying a system of nonredundant coordinates at any given step. As discussed in the next subsection and in Appendix 3.1, the dependence of  $\mathbf{V}$  on molecular geometry will be approximately accounted for in the quasi-Newton updating procedure; it is only when (a) we are initializing the Hessian and (b) when we are adding new coordinates to the Hessian (cf. Section 3.3.D) that Eq. (3.35) is used.

# G. Updating procedure for the Hessian matrix: The secant condition in vcoordinates.

As in other quasi-Newton methods, in our method the Hessian is updated using the differences in gradient-calculations from previous iterations. Since our optimization procedure works in the v-coordinates, we need to derive a secant condition for updating  $\mathbf{H}_{\nu}$ . As explained in Appendix 3.1, this is not entirely straightforward because V depends on the molecular geometry. In our work, we have selected the secant condition,

$$\mathbf{H}_{\nu}^{\text{old}} \boldsymbol{\delta} \mathbf{v} \approx \boldsymbol{\delta} \mathbf{g}_{\nu} - \left(\mathbf{V}^{\text{old}}\right)^{T} \left(\mathbf{B}^{\text{old}}\right)^{+} \left(\left(\mathbf{B}^{\text{old}}\right)^{T} \boldsymbol{\delta} \mathbf{V} \mathbf{g}_{\nu}^{\text{old}} + \left(\boldsymbol{\delta} \mathbf{B}\right)^{T} \mathbf{g}_{q}^{\text{old}}\right), \quad (3.40)$$

because we judged it to be among the simplest of the reasonable choices.

#### H. Updating procedure for the Hessian matrix: Finite differences

Our program is based on the idea that the key to effective transition-state optimization is to provide high accuracy for the Hessian elements associated with the key chemical modes, the reduced coordinates  $\{\mathbf{v}^{(r)}\}_{r=1}^{R}$ . (This idea was originally explored in

ref. 37.) We therefore use finite differences to approximate the first *R* rows (and columns, by symmetry) of the **v**-coordinate Hessian. The finite-difference approximation for the  $r^{\text{th}}$ row/column of the Hessian is given by the secant update formula, (3.40), using the perturbation  $\delta \mathbf{v} = \varepsilon \mathbf{e}_r$ , where  $\mathbf{e}_r$  is a unit vector that is zero except for the value one in the  $r^{\text{th}}$  position. Specifically,

$$\mathbf{H}\mathbf{e}_{r} = \frac{d\mathbf{g}_{v}}{d\varepsilon} - \mathbf{V}^{T}\mathbf{B}^{+} \left(\mathbf{B}^{T}\frac{d\mathbf{V}}{d\varepsilon}\mathbf{g}_{v} + \left(\frac{d\mathbf{B}}{d\varepsilon}\right)^{T}\mathbf{g}_{q}\right) \qquad r = 1, 2, \dots R$$
(3.41)

where the matrices and vectors on the right-hand-side of the equation are computed at the unperturbed geometry, and all the derivatives that appear are approximated with the finite difference approximation,

$$\frac{d\mathbf{f}\left(\mathbf{v}+\varepsilon\mathbf{e}_{r}\right)}{d\varepsilon} = \frac{\mathbf{f}\left(\mathbf{v}+\varepsilon\mathbf{e}_{r}\right)-\mathbf{f}\left(\mathbf{v}\right)}{\varepsilon}.$$
(3.42)

The results we present in this chapter use  $\varepsilon = 0.001$ .

Because the elements of the Hessian corresponding to the reduced coordinates are often quite accurate, it is inefficient, and unnecessary, to update the Hessian with finitedifferences at each iteration. Therefore, we only update the Hessian with finite differences for reduced coordinates if

$$\left|g_{\nu}^{(r)}\right| > \omega \frac{\left|\mathbf{g}_{\nu}\right|}{\sqrt{3N_{\text{atoms}} - 6(5)}} \tag{3.43}$$

and

$$\left|\mathbf{H}_{\nu}^{(k)}\mathbf{e}_{r}-\mathbf{H}_{\nu}^{(k-1)}\mathbf{e}_{r}\right| > \nu \left|\mathbf{H}_{\nu}^{(k-1)}\mathbf{e}_{r}\right|$$
(3.44)

where  $\mathbf{H}_{v}^{(k+1)}$  and  $\mathbf{H}_{v}^{(k)}$  are the Hessian approximations for the next iteration and from the present iteration, respectively. The user parameters  $\omega$  and v should be optimized, but we have not done this because the  $\omega = v = 1.0$  has worked satisfactorily in all our tests.

Criterion (3.43) is important because if the gradient in a specific direction is lower than the root-mean-square gradient, then the optimization in that direction is not hindering the overall rate of convergence, and so no intervention is needed. Criterion (3.44) reflects the fact that when the quasi-Newton update for a row of the Hessian is relatively small, the quasi-Newton update for that row of the Hessian is probably sufficiently accurate, and the computationally expensive finite-difference approximation can be avoided. With these criteria and reasonable values for the parameters  $\omega$  and v, we notice that finite-difference updates are relatively common during the first few iterations, but very rare after that.

# I. Updating procedure for the Hessian matrix: Quasi-Newton methods

After each new gradient calculation, the v-coordinate Hessian is updated using a quasi-Newton updating formula. The ingredients in this update are the last step taken by the algorithm,

$$\mathbf{s}_{v}^{(k)} = \mathbf{v}^{(k+1)} - \mathbf{v}^{(k)} \tag{3.45}$$

and the secant condition for updating the Hessian,

$$\mathbf{y}_{\nu}^{(k)} = \left(\mathbf{g}_{\nu}^{(k+1)} - \mathbf{g}_{\nu}^{(k)}\right) - \left(\mathbf{V}^{(k+1)}\right)^{T} \left(\mathbf{B}^{(k+1)}\right)^{+} \left(\left(\mathbf{B}^{(k+1)}\right)^{T} \left(\mathbf{V}^{(k+1)} - \mathbf{V}^{(k)}\right) \mathbf{g}_{\nu}^{(k+1)}\right) + \left(\mathbf{B}^{(k+1)} - \mathbf{B}^{(k)}\right)^{T} \mathbf{g}_{q}^{(k+1)}\right)$$
(3.46)

We have considered more than 100 different quasi-Newton updates; results of those comparisons will be reported separately. In this chapter, we consider only four updates: the simple-rank-one update  $(SR1)^{69,70}$ 

$$\mathbf{H}_{\nu}^{(k+1)} = \begin{cases} \mathbf{H}_{\nu}^{(k)} & \frac{\left(\left(\mathbf{y}_{\nu}^{(k)} - \mathbf{H}_{\nu}^{(k)}\mathbf{s}_{\nu}^{(k)}\right) \cdot \mathbf{s}_{\nu}^{(k)}\right)^{2}}{\left\|\mathbf{y}_{\nu}^{(k)} - \mathbf{H}_{\nu}^{(k)}\mathbf{s}_{\nu}^{(k)}\right\|^{2} \left\|\mathbf{s}_{\nu}^{(k)}\right\|^{2}} \leq 1 \times 10^{-18} \\ \mathbf{H}_{\nu}^{(k)} + \frac{\left(\mathbf{y}_{\nu}^{(k)} - \mathbf{H}_{\nu}^{(k)}\mathbf{s}_{\nu}^{(k)}\right)\left(\mathbf{y}_{\nu}^{(k)} - \mathbf{H}_{\nu}^{(k)}\mathbf{s}_{\nu}^{(k)}\right)^{T}}{\left(\mathbf{y}_{\nu}^{(k)} - \mathbf{H}_{\nu}^{(k)}\mathbf{s}_{\nu}^{(k)}\right) \cdot \mathbf{s}_{\nu}^{(k)}} & \text{otherwise} \end{cases}$$

$$(3.47)$$

the Powell-symmetric-Broyden update (PSB)<sup>70,71</sup>

$$\mathbf{H}_{v}^{(k+1)} = \mathbf{H}_{v}^{(k)} + \frac{\left(\mathbf{y}_{v}^{(k)} - \mathbf{H}_{v}^{(k)}\mathbf{s}_{v}^{(k)}\right)\left(\mathbf{s}_{v}^{(k)}\right)^{T} + \mathbf{s}_{v}^{(k)}\left(\mathbf{y}_{v}^{(k)} - \mathbf{H}_{v}^{(k)}\mathbf{s}_{v}^{(k)}\right)^{T}}{\left(\mathbf{s}_{v}^{(k)}\right)^{T}\mathbf{s}_{v}^{(k)}} - \left(\frac{\left(\mathbf{s}_{v}^{(k)}\right)^{T}\left(\mathbf{y}_{v}^{(k)} - \mathbf{H}_{v}^{(k)}\mathbf{s}_{v}^{(k)}\right)}{\left(\left(\mathbf{s}_{v}^{(k)}\right)^{T}\mathbf{s}_{v}^{(k)}\right)^{2}}\right)\mathbf{s}_{v}^{(k)}\left(\mathbf{s}_{v}^{(k)}\right)^{T}$$
(3.48)

the Broyden-Fletcher-Goldfarb-Shanno update (BFGS)<sup>72-76</sup>

$$\mathbf{H}_{v}^{(k+1)} = \mathbf{H}_{v}^{(k)} + \frac{\mathbf{y}_{v}^{(k)} \left(\mathbf{y}_{v}^{(k)}\right)^{T}}{\left(\mathbf{y}_{v}^{(k)}\right)^{T} \mathbf{s}_{v}^{(k)}} - \frac{\left(\mathbf{H}_{v}^{(k)} \mathbf{s}_{v}^{(k)}\right) \left(\mathbf{H}_{v}^{(k)} \mathbf{s}_{v}^{(k)}\right)^{T}}{\left(\mathbf{s}_{v}^{(k)}\right)^{T} \mathbf{H}_{v}^{(k)} \mathbf{s}_{v}^{(k)}}$$
(3.49)

and Bofill's 1994 update (Bofill),<sup>77</sup> which mixes the SR1 and PSB updates

$$\mathbf{H}_{\text{Bofill}}^{(k+1)} = (1 - \boldsymbol{\psi}) \mathbf{H}_{\text{SR1}}^{(k+1)} + \boldsymbol{\psi} \mathbf{H}_{\text{PSB}}^{(k+1)}$$
(3.50)

$$\boldsymbol{\psi} = 1 - \frac{\left| \mathbf{s}_{\nu}^{(k)} \cdot \left( \mathbf{y}_{\nu}^{(k)} - \mathbf{H}_{\nu}^{(k)} \mathbf{s}_{\nu}^{(k)} \right) \right|^{2}}{\left| \mathbf{y}_{\nu}^{(k)} - \mathbf{H}_{\nu}^{(k)} \mathbf{s}_{\nu}^{(k)} \right|^{2}} = \frac{\left| \mathbf{s}_{\nu}^{(k)} \times \left( \mathbf{y}_{\nu}^{(k)} - \mathbf{H}_{\nu}^{(k)} \mathbf{s}_{\nu}^{(k)} \right) \right|^{2}}{\left| \mathbf{s}_{\nu}^{(k)} \right|^{2} \left| \mathbf{y}_{\nu}^{(k)} - \mathbf{H}_{\nu}^{(k)} \mathbf{s}_{\nu}^{(k)} \right|^{2}}$$
(3.51)

 $\psi$  is the square of the sine of the angle between the step,  $\mathbf{s}_{v}^{(k+1)}$ , and the error in the old Hessian's approximation to the change in gradient that accompanies the step. The form of the SR1 update in Eq. (3.47) is designed to avoid divide-by-zero errors.

The SR1 and PSB updates do not preserve positive (semi)definiteness in the Hessian approximations; they are therefore appropriate for finding transition states.<sup>77</sup> The SR1 update, in particular, has the desirable characteristic of leading to very accurate approximations to the Hessian,<sup>78</sup> although it tends to converge more slowly than rank-two updates. The Bofill update is a good compromise, performing better than either PSB or SR1.<sup>77</sup>

The BFGS update maintains positive (semi)definiteness of the Hessian. Moreover, the mathematical derivation of the BFGS quasi-Newton update is invalid when the curvature condition,  $\mathbf{y}_{v}^{(k)} \cdot \mathbf{s}_{v}^{(k)} > 0$  is violated and the BFGS update is numerically illconditioned when  $\mathbf{y}_{\nu}^{(k)} \cdot \mathbf{s}_{\nu}^{(k)} \approx 0$ . The curvature condition and positive-definiteness of the Hessian is maintained in the damped-BFGS procedure.<sup>79,80</sup> For transition-state optimization, however, we wish to allow violations of the curvature condition for motions associated with the transition-state. We therefore introduce a new damped-BFGS procedure, where only the non-reduced coordinates are included in the damping procedure. The damped BFGS update defines an effective **y**-vector,

$$\mathbf{r}_{v}^{(k)} = \mathbf{\Theta} \mathbf{y}_{v}^{(k)} + (\mathbf{I} - \mathbf{\Theta}) \mathbf{H}_{v}^{(k)} \mathbf{s}_{v}^{(k)}$$
(3.52)

in terms of the diagonal matrix  $\Theta_{ij} = \delta_{ij}\theta_i$ . The elements corresponding to the reduced coordinates are never damped,

$$\theta_i = 1$$
  $i = 1, 2, \dots, R$  (3.53)

The non-reduced coordinates, i > R are damped according to the formula

$$\boldsymbol{\theta}_{i>R} = \begin{cases} 1 & \sum_{m=R+1}^{3N_{\text{atoms}}-6(5)} s_m^{(k)} y_m^{(k)} \ge (0.2) \sum_{m,n=R+1}^{3N_{\text{atoms}}-6(5)} s_m^{(k)} h_{mn}^{(k)} s_n^{(k)} \\ \frac{(0.8) \sum_{m,n=R+1}^{3N_{\text{atoms}}-6(5)} s_m^{(k)} h_{mn}^{(k)} s_n^{(k)}}{\sum_{m,n=R+1}^{3N_{\text{atoms}}-6(5)} s_m^{(k)} h_{mn}^{(k)} s_n^{(k)}} & \sum_{m=R+1}^{3N_{\text{atoms}}-6(5)} s_m^{(k)} y_m^{(k)} \le (0.2) \sum_{m,n=R+1}^{3N_{\text{atoms}}-6(5)} s_m^{(k)} h_{mn}^{(k)} s_n^{(k)} \end{cases}$$

$$(3.54)$$

The damped-BFGS update is obtained by replacing  $\mathbf{y}_{v}^{(k)}$  with  $\mathbf{r}_{v}^{(k)}$  in the BFGS update formula, Eq. (3.49).

Especially when using updates based on the BFGS formula, we observed that the quasi-Newton update often caused large changes in elements of the Hessian corresponding to the reduced coordinates. We wish to prevent the quasi-Newton updates from making large changes to these elements of the Hessian, which have usually been

fairly accurately determined by the finite-difference procedure. Accordingly, the row/column of the Hessian corresponding to the reduced subspace is only updated if the following condition is satisfied:

$$\left| \mathbf{H}_{\nu}^{(k)} \mathbf{e}_{r} - \mathbf{H}_{\nu}^{(k-1)} \mathbf{e}_{r} \right| < \kappa \left| \mathbf{H}_{\nu}^{(k-1)} \mathbf{e}_{r} \right| \qquad r = 1, 2, \dots, R$$
(3.55)

Optimal values for the user-parameter  $\kappa$  should be determined, but we have not done this. However, choosing  $\kappa = 1$  has sufficed for our tests.

#### J. Updating procedure for the Hessian matrix: Eigenstructure Modification

At the transition state, the Hessian should have exactly one negative eigenvalue. Moreover, we expect that the eigenvector that corresponds to that eigenvalue will feature motions of the reduced coordinates, as these are the key coordinates associated with the chemical change. In cases where the approximate Hessian matrix does not have this structure, either because the molecular structure under consideration is far from the transition state or because the Hessian is inaccurate, we modify the Hessian to force this structure. (This revision is similar, conceptually, to eigenvector-following methods.<sup>81-84</sup>)

First we ensure that the non-reduced block of the Hessian, is positive semidefinite. If the non-reduced block of the Hessian is not positive semidefinite, we replace it by the nearest positive semidefinite matrix. This is achieved by diagonalizing the non-reduced Hessian, setting all negative eigenvalues to zero, and then reconstructing the non-reduced Hessian.

Second, we ensure that the reduced block of the Hessian has one, sufficiently negative, eigenvalue. The reduced-Hessian is diagonalized, and if the smallest eigenvalue

is larger than a threshold,  $\lambda_n$ , we reset the eigenvalue to that threshold. (We have not optimized the choice of this parameter, instead opting for the value  $\lambda_n = -0.005$  a.u., which we believe to be reasonable.) If the reduced-Hessian has more than one negative eigenvalue, all but the smallest (most negative) eigenvalues are set equal to zero. The reduced Hessian therefore has exactly one negative frequency, and that frequency is less than or equal to  $\lambda_n$ .

We then diagonalize the entire Hessian matrix and examine its eigenstructure,

$$\mathbf{H}_{\nu}^{(k)} = \sum_{i=1}^{3N-6(5)} \lambda_i \boldsymbol{\chi}_i \boldsymbol{\chi}_i^T, \qquad (3.56)$$

where it is assumed that the eigenvalues,  $\lambda_i$  and their associated normalized eigenvectors,  $\xi_i$ , are listed in nondecreasing order,  $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_{3N_{\text{atoms}}-6(5)}$ . We wish to have one eigenvalue that is less than or equal to  $\lambda_n$ , and  $3N_{\text{atoms}} - 7(6)$  eigenvalues that are greater than or equal to a positive threshold,  $\lambda_p$ . (We have chosen  $\lambda_p = 0.005$  for our numerical tests.) I.e., the correct eigenstructure is

$$\lambda_{1} \leq \lambda_{n}$$

$$\lambda_{i} \geq \lambda_{p} \qquad \qquad i = 2, 3, \dots 3N_{\text{atoms}} - 6(5) \qquad (3.57)$$

In most cases we observe that the Hessian matrix has an acceptable eigenstructure.

If the eigenvalues of the Hessian do not satisfy Eq. (3.57), but the Hessian has one negative eigenvalue, we modify the eigenstructure by making the replacements

$$\lambda_{i} = \min(\lambda_{i}, \lambda_{n})$$

$$\lambda_{i} = \max(\lambda_{i}, \lambda_{p}) \qquad i = 2, 3, \dots, 3N_{\text{atoms}} - 6(5)$$
(3.58)

in Eq. (3.56).

If the Hessian has multiple negative eigenvalues, we need to select one as the candidate for the negative-curvature direction. To do this, for each eigenvector with a negative eigenvalue, we compute the fraction of the eigenvector that resides in the reduced space,

$$p_{i} = \sum_{r=1}^{R} \left| \chi_{i;r} \right|^{2}$$
(3.59)

The eigenvector that has the largest value of  $p_i$  is chosen to retain a negative eigenvalue; if its eigenvalue is larger than the negative threshold, then its eigenvalue is replaced by  $\lambda_n$ . The other eigenvectors will have their corresponding eigenvalues changed to max $(\lambda_p, \lambda_i)$ .

If the Hessian has no negative eigenvalues, we compute the fraction of each eigenvector that lies in the reduced space using Eq. (3.59). Among the eigenvectors that are predominately in the reduced space ( $p_i \ge 0.5$ ), we select the one with the smallest eigenvalue. The eigenvalue of this eigenvector is set equal to the negative threshold; the other eigenvalues are forced to be no smaller than the positive threshold.

#### K. Step direction: Scaled-Newton or rational function step?

Given an (approximate) Hessian matrix with the correct eigenstructure and the gradient of the molecular potential energy for a particular molecular geometry, Newton's method predicts that the transition state—the geometry that has zero gradient—is located at  $\mathbf{v}^{(k+1)} = \mathbf{v}^{(k)} + \mathbf{s}^{(k)}_{v}$ , where  $\mathbf{s}^{(k)}_{v}$  is obtained by solving the linear equations,

$$\mathbf{H}_{\nu}^{(k)}\mathbf{s}_{\nu}^{(k)} = -\mathbf{g}_{\nu}^{(k)} \tag{3.60}$$

In terms of the eigenvectors and eigenvalues of the Hessian, the Newton step can be expressed as

$$\mathbf{s}_{v}^{(k)} = -\sum_{i=1}^{3N-6(5)} \left( \frac{\mathbf{\chi}_{i}^{T} \mathbf{g}_{v}^{(k)}}{\lambda_{i}} \right) \mathbf{\chi}_{i}$$
(3.61)

Taking the Newton step would lead to the exact transition-state if (a) the Hessian were computed exactly, instead of approximately and (b) cubic and higher-order terms in the Taylor expansion of the molecular potential energy surface were negligible. Neither of these assumptions is true. The cubic and higher-order terms are negligible, however, if the stepsize,  $|\mathbf{s}_{\nu}^{(k)}|$ , is short enough. This motivates the idea of a trust radius: one uses the step from Eq. (3.60) as long as the step lies within a spherical region, defined by the trust radius,  $\tau$ , centered on the current molecular conformation. When the stepsize obtained from Eq. (3.60) is larger than the trust radius, the step should not be trusted because the quadratic approximation to the molecular potential energy surface is not reliable for steps that large. Instead, we should make the best-possible-step with stepsize equal to the trust radius. We considered two different definitions for the best-possible-step.

#### 1. Trust-region image potential (TRIM) (Scaled Newton step)

The trust-region image potential (TRIM) step is obtained by increasing the magnitude of the eigenvalues by the amount  $\tilde{\lambda} \ge 0$ ,<sup>85,86</sup>

$$\mathbf{s}_{\nu}^{(k)}\left(\tilde{\lambda}\right) = -\left(\frac{\mathbf{\chi}_{1}^{T}\mathbf{g}_{\nu}^{(k)}}{\lambda_{1}-\tilde{\lambda}}\right)\mathbf{\chi}_{1} - \sum_{i=2}^{3N-6(5)}\left(\frac{\mathbf{\chi}_{i}^{T}\mathbf{g}_{\nu}^{(k)}}{\lambda_{i}+\tilde{\lambda}}\right)\mathbf{\chi}_{i}$$
(3.62)

until the stepsize matches the trust radius,

$$\left|\mathbf{s}_{v}^{(k)}\left(\tilde{\lambda}\right)\right|^{2} = \tau^{2}$$
(3.63)

Eq. (3.63) is a one-dimensional nonlinear equation which is easy to solve using a method like Ridder's method.<sup>87</sup>

### 2. Rational function optimization (RFO)

The rational function optimization (RFO) step uses a Pade approximant to estimate the effects of higher-order terms in the Taylor series.<sup>38,77,86,88-91</sup> It is a refinement of Eq. (3.62) because it uses separate shifts for the negative and positive eigenvalues,

$$\mathbf{s}_{\nu}^{(k)}(\boldsymbol{\varsigma}) = -\left(\frac{\mathbf{\chi}_{1}^{T}\mathbf{g}_{\nu}^{(k)}}{\lambda_{1} - \tilde{\lambda}_{p}(\boldsymbol{\varsigma})}\right)\mathbf{\chi}_{1} - \sum_{i=2}^{3N-6(5)}\left(\frac{\mathbf{\chi}_{i}^{T}\mathbf{g}_{\nu}^{(k)}}{\lambda_{i} + \tilde{\lambda}_{n}(\boldsymbol{\varsigma})}\right)\mathbf{\chi}_{i}$$
(3.64)

The eigenvalue-shift parameters are obtained by solving the generalized eigenvalue problems

$$\begin{bmatrix} \lambda_{1} & \chi_{1}^{T} \mathbf{g} \\ \mathbf{g}^{T} \boldsymbol{\chi}_{1} & 0 \end{bmatrix}^{\mathbf{u}_{j}^{(-)}} = \lambda_{j}^{(-)} \begin{bmatrix} \boldsymbol{\varsigma} & 0 \\ 0 & 1 \end{bmatrix}^{\mathbf{u}_{j}^{(-)}}$$
(3.65)  
$$\begin{bmatrix} \lambda_{2} & 0 & 0 & \cdots & \boldsymbol{\chi}_{2}^{T} \mathbf{g} \\ 0 & \lambda_{3} & 0 & \cdots & \boldsymbol{\chi}_{3}^{T} \mathbf{g} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_{3N-6} & \boldsymbol{\chi}_{3N-6}^{T} \mathbf{g} \end{bmatrix}^{\mathbf{u}_{j}^{(+)}} = \lambda_{j}^{(+)} \begin{bmatrix} \boldsymbol{\varsigma} & 0 & 0 & \cdots & 0 \\ 0 & \boldsymbol{\varsigma} & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{\varsigma} & 0 \\ 0 & 0 & \cdots & 0 & 1 \end{bmatrix}^{\mathbf{v}_{j}^{(+)}}$$
(3.65)

(This assumes a diagonal scaling matrix; this is not essential but the mathematics is more difficult in other cases.) The shift-parameter for the negative curvature mode is the highest eigenvalue of Eq. (3.65);  $\tilde{\lambda}_n(\varsigma) = \max(\lambda_j^{(-)}) > 0$ ; the shift parameter for the positive-curvature modes is minus the lowest eigenvalue of Eq. (3.66),  $\tilde{\lambda}_p(\varsigma) = -\min(\lambda_j^{(+)}) > 0$ . The value of  $\varsigma$  is determined by requiring that the stepsize is equal to the trust radius, as in Eq. (3.63).

# L. Step-size determination: Trust radius methods

The length of the steps computed by TRIM or RFO are limited by a trust radius, which should be chosen to be large enough for rapid convergence but small enough to ensure the accuracy of the quadratic approximation to the potential energy surface. In order to ensure that the step never gets too large or too small, we define a minimum and maximum trust radius,

$$\tau_{\rm max} = \sqrt{N_{\rm atoms}} \, \text{a.u.} \tag{3.67}$$

$$\tau_{\min} = \frac{1}{10} \sqrt{N_{\text{atoms}}} \quad \text{a.u.} \tag{3.68}$$

Our initial trust radius was  $\tau_{\text{initial}} = .35\sqrt{N_{\text{atoms}}}$  a.u.

In our approach, we start by comparing the Cartesian gradient before and after the step. If the magnitude of the gradient has decreased,  $|\mathbf{g}_{x}^{(k+1)}| < |\mathbf{g}_{x}^{(k)}|$ , then we accept the proposed step and update the trust radius using either (a) the energy-based criterion or (b) the gradient-based criteria described in the following paragraphs. If the magnitude of the

gradient increases, we decrease the trust radius by a factor of four,  $\tau_{new} = \tau_{old}/4$ . We then compute a step with this new shorter length, hoping that the magnitude of the gradient will decrease. We do not wish to allow steps that are too short, because we strongly believe that the quadratic model is accurate when none of the Cartesian coordinates change by more than about .1 a.u.. Therefore, if  $\tau_{new} < \tau_{min}/10$ , we set  $\tau_{new} = \tau_{min}$  and *take the step whether the magnitude of the gradient increases or not*. This last criterion is very useful when the initial guess for the transition-state structure is poor. If the current molecular structure lies in a region of the potential energy surface where all the frequencies are positive, a short uphill step in the direction of the transition state will cause the norm of the gradient to increase. The preceding method ensures that such uphill steps are allowed.

Our program includes two different approaches for updating the trust radius.

#### 1. Energy-based trust radius updating

The predicted change in energy between the new geometry and the previous geometry is computed using the quadratic approximation to the energy in **v**-space,

$$\Delta m^{(k)} = \mathbf{g}_{\nu}^{(k)} \cdot \mathbf{s}_{\nu}^{(k)} + \frac{1}{2} \left( \mathbf{s}_{\nu}^{(k)} \right)^{T} \mathbf{H}_{\nu}^{(k)} \mathbf{s}_{\nu}^{(k)}$$
(3.69)

This change is compared to the actual change in energy,  $\Delta U^{(k)} = U(\mathbf{x}^{(k+1)}) - U(\mathbf{x}^{(k)})$ . If the quadratic approximation is accurate, the actual change in energy should be very close to the change predicted by the quadratic model. If

$$\frac{2}{3} < \frac{\Delta m^{(k)}}{\Delta U^{(k)}} < \frac{3}{2}, \tag{3.70}$$

then the quadratic model for the molecular potential energy is highly accurate and we double the trust radius,  $\tau_{new} = \min(\max(2\tau_{old}, \tau_{min}), \tau_{max})$ . The inner maximum in this criterion is needed because sometimes the trust radius that has been obtained after repeated rejection of a step is much less than  $\tau_{min}$ . If

$$\frac{1}{3} < \frac{\Delta m^{(k)}}{\Delta U^{(k)}} < 3, \qquad (3.71)$$

then the quadratic model is of moderate accuracy and we retain the trust radius,  $\tau_{\text{new}} = \max(\tau_{\text{old}}, \tau_{\min})$ . Otherwise, the quadratic model is inaccurate for steps as large as the one we just accepted, so we reduce the trust radius by a factor of four,  $\tau_{\text{new}} = \min(\frac{1}{4}\tau_{\text{old}}, \tau_{\min})$ . Notice that when the quadratic model predicts a change in energy that differs in sign from the true change in energy, the trust radius is always reduced.

#### 2. Gradient-based trust radius updating

The energy-based trust radius updating scheme used in the preceding section is traditional,<sup>37,80</sup> and it is clearly appropriate for minimization of the energy. Such an approach is less clearly appropriate when the objective is to find a saddle point; in saddle-point problems, it seems more natural to update the trust radius based on the accuracy of the predicted,

$$\mathbf{g}_{\nu;\text{predicted}}^{(k+1)} = \mathbf{g}_{\nu}^{(k)} + \mathbf{H}_{\nu}^{(k)} \mathbf{s}_{\nu}^{(k)} .$$
(3.72)

Specifically, we will adjust the trust radius based on (1) whether the change in the magnitude of the gradient that is predicted resembles the observed change in magnitude of the gradient, as measured by the ratio

$$\rho = \frac{\left| \mathbf{g}_{\nu; \text{predicted}}^{(k+1)} \right| - \left| \mathbf{g}_{\nu}^{(k)} \right|}{\left| \mathbf{g}_{\nu}^{(k+1)} \right| - \left| \mathbf{g}_{\nu}^{(k)} \right|}$$
(3.73)

and (2) whether the change in direction of gradient that is predicted is aligned with the change in gradient direction that is observed, as measured by the cosine,

$$\cos\left(\theta\right) = \frac{\left(\mathbf{g}_{\nu;\text{predicted}}^{(k+1)} - \mathbf{g}_{\nu}^{(k)}\right) \cdot \left(\mathbf{g}_{\nu}^{(k+1)} - \mathbf{g}_{\nu}^{(k)}\right)}{\left|\mathbf{g}_{\nu;\text{predicted}}^{(k+1)} - \mathbf{g}_{\nu}^{(k)}\right| \cdot \left|\mathbf{g}_{\nu}^{(k+1)} - \mathbf{g}_{\nu}^{(k)}\right|}$$
(3.74)

As the dimension increases, the chance of two vectors having a cosine close to one decreases: it is much more rare for random vectors in high-dimensional spaces to be aligned than it is for vectors in low-dimensional spaces. For example, if one generates a large number of random vectors in d dimensions, then 10% of them will satisfy

$$\cos(\theta) \ge p_{10}(d) \approx \sqrt{\frac{1.6424}{d} + \frac{1.11}{d^2}}$$
 (3.75)

and 40% of them will satisfy

$$\cos(\theta) \ge p_{40}(d) \approx \sqrt{\frac{0.064175}{d} + \frac{0.0946}{d^2}}.$$
 (3.76)

The approximate expressions above were derived by least-squares fitting to much more complicated analytical expressions, and are quite accurate for vectors with ten or more dimensions. The complete mathematical treatment of the gradient-based trust-radius update is quite involved, and will be published separately.

When a step is accepted, we compute Eqs. (3.73) and (3.74). If

$$\frac{\frac{4}{5} < \rho < \frac{5}{4}}{p_{10} \left( 3N_{\text{atoms}} - 6(5) \right) < \cos(\theta)}$$
(3.77)

then the quadratic model is predicting accurate gradients, and the trust radius is doubled,  $\tau_{\text{new}} = \min(\max(2\tau_{\text{old}}, \tau_{\min}), \tau_{\max}).$ 

$$\frac{\frac{1}{5} < \rho < 6}{p_{40} \left(3N_{\text{atoms}} - 6(5)\right) < \cos(\theta)}$$
(3.78)

then the approximate gradient was of moderate accuracy, and the trust radius is retained,  $\tau_{\text{new}} = \max(\tau_{\text{old}}, \tau_{\min})$ . Otherwise the last step went outside the region where the gradient approximated using Eq. (3.72) is inaccurate, so we halve the trust radius,  $\tau_{\text{new}} = \min(\frac{1}{2}\tau_{\text{old}}, \tau_{\min})$ .

# 3. Summary

Our trust-radius algorithm can be summarized in two steps:

- 1. If  $|\mathbf{g}_x^{(k+1)}| < |\mathbf{g}_x^{(k)}|$ , accept step and update the trust radius using either (a) the energy-based method or (b) the gradient-based method.
- 2. Otherwise, change the trust radius to  $\tau_{new} = \frac{1}{4}\tau_{old}$ . If  $\tau_{new} \ge \frac{1}{10}\tau_{min}$ , attempt a new step with this length and go back to step 1. Otherwise, set  $\tau_{new} = \tau_{min}$  and *always accept this step*.

#### M. Convergence Criteria

The program is flexible with respect to stopping conditions. In our studies, we have used the protocol proposed by Baker and Chan.<sup>42</sup> Convergence is considered to have been achieved if (a) the largest component of the Cartesian gradient is less than  $3.0 \times 10^{-4}$  a.u. and (b) either the energy change in the last iteration is less than  $1.0 \times 10^{-6}$  a.u. or the largest component of the step (in Cartesian coordinates) is less than  $3.0 \times 10^{-4}$  a.u.. If the method does not converge in 200 iterations, then we consider this to be a convergence failure. We observed that calculations that fail to converge in 200 iterations rarely converge even in 1000 iterations and, even if they do eventually converge, rarely converge to the targeted transition state.

# 3.4 Overview of Saddle

Putting together the methods described in the previous section, the final *Saddle* algorithm follows. Steps **i** to **iv** are initialization steps. Steps **v** is the Hessian-modification step, where finite-difference approximations to the key "reduced" coordinates are made (if necessary) and where the Hessian is modified to have a desirable eigenstructure. Step **vi** is where the step is computed. Steps **vii** to **xi** are associated with accepting/rejecting the step, updating the trust radius, the quasi-Newton update to the Hessian, and other preliminaries that are necessary to prepare for the next step.

i. The input is the initial structure in Cartesian coordinates, the initial Hessian, and an initial trust radius.

- **ii.** Select a system of redundant coordinates. If the user does not specify reduced coordinates, select reduced coordinates automatically. In the calculations we report, the reduced coordinates were always manually specified.
- iii. Calculate the molecular potential energy and its gradient using an external program/subroutine. (We are using *Gaussian* '09, but the program is flexible.)
  Compute the Wilson B matrix, Eq. (3.4), and transform initial Hessian and gradient to the internal space using equations (3.13) and (3.14).
- iv. Construct the V matrix and transform the gradient and Hessian to v-space using equations (3.33) and (3.35).
- v. Check to see if finite-difference updates to the elements of the Hessian involving the reduced coordinates are needed. In the first step, the user has the option to skip finite-difference updates (this is useful if the exact, or an accurate, initial Hessian is used) or to use only the gradient condition, Eq. (3.43). In subsequent steps, finite-difference updates are performed only if both criteria (3.43) and (3.44) are satisfied. Modify the eigenstructure of the v-space Hessian, as described in section 3.3.J.
- vi. Diagonalize the Hessian and take a step with length given by the trust radius, using either the scaled Newton step (i.e., trust region image method, TRIM; Eq. (3.62)) or the step from the rational function optimization procedure (RFO; Eq. (3.64)).
- vii. Express the step in internal coordinates, Eq. (3.38), and then convert the step in internal coordinates to Cartesian coordinates using the manifold-projection

method, Eq. (3.17). Calculate the molecular potential energy and its gradient using an external program/subroutine.

- viii. If the magnitude of the gradient decreased, accept the step. Otherwise, decrease the trust radius and go back to vi to compute a shorter step. If the trust radius is less than  $\tau_{min}/10$ , a step of length  $\tau_{min}$  is taken, and automatically accepted.
- ix. Now that the step is accepted, update the trust radius using either the energy-based (section **3.3.L.1**) or gradient-based (section **3.3.L.2**) procedures.
- **x.** Construct a new **v**-space, and transform the gradient and step vector to the new **v**-space.
- xi. Update the Hessian in the new v-space using one of the methods in section 3.3.I.
- xii. Check to see if the convergence criteria, section 3.3.M, are satisfied. If the calculation has not converged, check to see if new internal coordinates are present and, if so, add them and update the Hessian accordingly. Go back to step v to modify the Hessian in preparation for the next iteration.
### **3.5** Numerical Tests

#### A. Computational Methods

To assess the efficiency of the algorithm summarized in the previous section, we optimized the transition states for 20 chemical reactions. These reactions, together with the reduced coordinates we chose, are listed in appendix 3.2. All the calculations were performed using HF/3-21G(d); the quasi-Newton procedure was initialized by calculating the Hessian exactly at the initial geometry. *Gaussian '09* was used for the electronic structure calculations.<sup>41</sup>

# B. Assessment: Methods for step direction (TRIM, RFO) and step size (trust radius)

To test the different algorithmic approaches for controlling the step-size and stepdirection, we tested the proposed trust-radius methods (energy-based and gradient-based) with the TRIM (scaled-Newton) and rational function optimization (RFO) methods for computing the step. We used the Bofill-1994 update (Eq. (3.50)) for this test, updating the reduced coordinates only when the condition in Eq. (3.55) was satisfied. We compared these approaches to the popular *Berny* optimization algorithm,<sup>40</sup> as implemented in *Gaussian* '09, with the standard convergence criteria. (The *Gaussian* command was opt(ts, calcfc, noeigentest).)

The results of these tests are presented in Table 3.1. The TRIM step required, on average, 50% fewer iterations than the RFO step. The gradient-based trust radius required

slightly fewer iterations. We select the TRIM step with the gradient-based trust radius as a computationally efficient and robust transition-state optimization protocol.

### C. Assessment: Quasi-Newton methods

In table 3.2, we compare the performance of three quasi-Newton methods: Bofill's 1994 update (Eqs. (3.50)-(3.51)), the conventional BFGS method (Eq. (3.49)), and the damped-BFGS method (Eqs. (3.49), (3.52)-(3.54)). The BFGS method would usually be expected to fail, because it assumes that the Hessian is positive semidefinite. However, the key chemical blocks of the Hessian are updated with finite-differences, and therefore have the correct structure. In addition, the eigenstructure of the Hessian is modified (see section 3.3.J) to make it more appropriate for transition-states. Because of this, the BFGS method converges in most cases; this is especially true if large quasi-Newton updates to the reduced blocks of the Hessian are prevented by imposing the condition in Eq. (3.55). Our adaptation of the damped-BFGS method, using Eqs. (3.52)-(3.54), is further removed from the traditional positive-semidefinite BFGS update. It performed better than the traditional BFGS approach. However, the Bofill-1994 update, which is explicitly designed to be appropriate for transition states and which has no link to any positive-semidefinite Hessian updating procedure, is the best of all the methods we tested: it performs better for every reaction we tested. Because the Bofill update can preserve the negative-curvature directions of the Hessian, it is less important to prevent large revisions to the reduced elements of the Hessian in this case. Nonetheless, marginally better results are obtained when the condition in Eq. (3.55) was imposed. We conclude using the condition in Eq. (3.55) to prevent large updates to the reduced elements of the Hessian is often helpful (every method we considered converged for all of our test reactions when this condition was imposed) and almost never harmful.

### D. Assessment: Comparison to the Berny Optimizer

One of the best and most popular conventional transition-state optimizers is the Berny optimizer, as implemented in the Gaussian program. The data presented in Table 3.1 allows the reader to compare our preferred protocol-TRIM with the Bofill-1994 quasi-Newton update and condition (3.55) to prevent spurious updates of the key "reduced" elements of the Hessian-to the Berny optimizer. Our method is expected to be slower for the easy cases because some gradient calculations are used to refine the Hessian, instead of making steps towards the transition state. Table 3.1 confirms this trend: the *Berny* optimizer is never the most efficient method if it requires more than 15 gradient calculations. Our method uses additional information about the nature of the targeted transition state (specifically, information about which internal coordinates are implicated in the chemical reaction of interest) and is expected to be more robust. Table 3.1 confirms this trend also: the Berny optimizer fails in 3 cases, while our preferred method converges for all the reactions we tested. In reaction #1, the *Berny* optimizer converged to the wrong transition state after 109 iterations; our methods avoid this fate because the chemically important coordinates are singled out for special treatment. In reactions #17 and #19, the Berny optimizer crashed because of an error associated with problems in the internal coordinate system. The robust dihedral descriptors (Eqs. (3.2)-(3.3)) and manifold-projection method (Eq. (3.17)) used by *Saddle* avoid these problems.

These results suggested that we have designed a method with the properties we sought: it is very robust, with little additional computational cost beyond more conventional methods.

**Table 3.1.** The number of gradient evaluations required to converge the reactions in Appendix 3.2 using the HF/3-21G(d) method for different methods for computing the step-direction (cf. section 3.3.K) and controlling the step size with a trust radius (cf. section 3.3.L). The Bofill-1994 quasi-Newton update, with condition (3.55) imposed, was used. The average number of gradient evaluations *when the method converges to the correct transition state* is tabulated so that the computational cost of different methods can be compared. For each reaction, the result for the most efficient version(s) of our approaches is bold-faced. When the *Berny* transition-state optimization is more efficient than any of our approaches, its results are underlined.

Reaction	Trust Radius	Image Method	Rational Fun	ction Model	Berny
	Energy Trust Radius	Gradient Trust Radius	Energy Trust Radius	Gradient Trust Radius	Transitio n-State Optimiz er
1	9	9	14	14	fails
2	14	14	15	15	<u>6</u>
3	17	16	20	22	<u>13</u>
4	14	14	20	31	20
5	21	25	23	22	<u>13</u>
6	13	12	10	10	<u>7</u>
7	25	22	26	27	<u>15</u>
8	22	21	15	15	51
9	12	13	86	86	19
10	6	6	6	6	<u>4</u>
11	8	8	7	7	8
12	10	10	13	13	<u>8</u>
13	6	6	6	6	<u>3</u>
14	17	18	16	19	31
15	15	15	49	49	17
16	27	32	59	28	<u>13</u>
17	16	15	36	36	fails
18	27	17	21	17	<u>8</u>
19	18	19	17	20	fails
20	11	11	9	9	<u>7</u>
Avg. # of					
gradient					
evaluations	15.4	15.2	23.4	22.6	14.3
Failures	0	0	0	0	3

**Table 3.2.** The number of gradient evaluations required to converge the reactions in Appendix 3.2 using the HF/3-21G(d) method for different quasi-Newton methods from section 3.3.I, depending on whether condition (3.55), restricting quasi-Newton updates to reduced elements of the Hessian, is imposed. The gradient-based trust radius and the scaled-Newton (trust region image method, TRIM) were used. The average number of gradient evaluations *when the method converges to the correct transition state* is tabulated as a measure of computational cost. For each reaction, the result for the most efficient version(s) of our approaches is bold-faced.

Reaction	Во	fill	BF	FGS	damped-	BFGS
	Standard	Using	Standard	Using eq.	Standard	Using
		eq.		(3.55)		eq.
		(3.55)				(3.55)
1	9	9	10	10	10	10
2	14	14	40	28	191	29
3	16	16	fails	101	63	46
4	14	14	11	11	11	11
5	25	25	138	71	117	40
6	11	12	21	22	21	22
7	22	22	fails	78	77	69
8	21	21	fails	fails	fails	36
9	13	13	fails	16	fails	fails
10	6	6	15	23	15	23
11	8	8	8	8	8	8
12	10	10	56	26	43	40
13	6	6	8	8	8	8
14	18	18	44	fails	68	29
15	15	15	15	15	15	15
16	31	32	fails	fails	fails	fails
17	15	15	87	14	30	21
18	17	17	53	59	52	45
19	19	19	177	83	80	55
20	11	11	40	93	40	83
Avg. # of						
gradient						
evaluations	15.1	15.2	48.2	39.7	49.9	32.8
# of						
failures	0	0	5	3	3	2

### 3.6 Summary

We have presented a new approach for transition-state optimization where the chemically-important "reduced" coordinates are singled out for special treatment. Most importantly, elements of the Hessian involving the reduced coordinates are accurately computed using finite differences. This work is essentially an adaptation of the approach in ref. <sup>37</sup>, which took a similar approach using Cartesian coordinates, to redundant internal coordinates. The adaptation was much more difficult than we expected, and several new ideas were needed.

The introduction of robust dihedral descriptors and the manifold-project method gives us a failsafe way to convert between redundant internal coordinates and Cartesian coordinates. These developments are described in chapter 2. For computational efficiency, we need to avoid using finite-differences in each iteration; the criteria in Eqs. (3.43) and (3.44) are used to ensure this. We found that several finite-difference steps are used to update the Hessian in early iterations, but in later iterations (near convergence) finite-difference steps are rare. The condition in Eq. (3.55) prevents large changes to the reduced blocks of the Hessian from quasi-Newton update and is especially useful when the quasi-Newton update is poor. Most trust-radius updating methods are based on the accuracy of quadratic model for the energy, but it is the gradient that is being optimized in saddle-point problems. This led us to develop the gradient-based trust-radius update in section 3.3.L.2; for transition-state optimization, the gradient-based trust-radius scheme method outperforms methods based on the value of the objective function. Finally, we tested the trust-region image method (TRIM, scaled Newton) and the rational function

optimization (RFO) steps; TRIM outperforms RFO in our tests. We recommend using the Bofill-1994 quasi-Newton Hessian update, together with the TRIM step using a trust radius computed by the gradient-based approach.

In chapter 4 we will provide more systematic tests of this approach. Even from the limited data provided in this chapter, it seems clear that this new approach is more robust than traditional methods. The new method is more expensive, but the additional computational expense is small.

The additional computational cost is probably even less than one might assume: there are many user parameters in this model, associated with (a) when finite-difference updates to the Hessian are performed, (b) when the quasi-Newton update is allowed to modify reduced blocks of the Hessian, (c) how the eigenstructure of the Hessian is modified (especially threshold values on the smallest positive and largest negative eigenvalues of the Hessian), and (d) the trust-radius update and the criteria for accepting/rejecting steps. We have not systematically optimized the values of these parameters, or even performed a casual search for parameter-values associated with good performance. Instead, we have chosen what we felt were reasonable "round numbers" for the values of these parameters. In our future work, we hope to find better choices for the values for these parameters.

## Appendix 3.1. The secant condition in delocalized nonredundant internal coordinates.

There are several different ways to correct the naïve secant condition in v-coordinates,

$$\mathbf{H}_{v}\delta\mathbf{v}\approx\mathbf{g}_{v}\left(\mathbf{v}+\delta\mathbf{v}\right)-\mathbf{g}_{v}\left(\mathbf{v}\right)$$
(3.79)

to approximately account for the dependence of V on the molecular geometry. The method we use in the program is derived starting from the secant condition in Cartesian coordinates,

$$\mathbf{H}_{x}^{\text{old}} \delta \mathbf{x} \approx \delta \mathbf{g}_{x} \tag{3.80}$$

where "new" refers to quantities evaluated at the point  $\mathbf{x} + \delta \mathbf{x}$ , "old" refers to quantities evaluated at  $\mathbf{x}$ , and  $\delta \mathbf{f}$  denotes  $\mathbf{f}^{\text{new}} - \mathbf{f}^{\text{old}}$ . To obtain an expression containing the Hessian in **v**-coordinates, we use the chain rule for the second derivative of the molecular potential energy, U,

$$\begin{bmatrix} \mathbf{H}_{x}^{\text{old}} \delta \mathbf{x} \end{bmatrix}_{i} = \sum_{j=1}^{3N_{\text{atoms}}} \frac{\partial^{2} U(\mathbf{x}^{\text{old}})}{\partial x_{i} \partial x_{j}} \delta x_{j}$$

$$= \sum_{k,l=1}^{3N_{\text{atoms}}-6(5)} \frac{\partial^{2} U(\mathbf{v}^{\text{old}})}{\partial v^{(k)} \partial v^{(l)}} \frac{\partial v^{(k;\text{old})}}{\partial x_{i}} \begin{bmatrix} \sum_{j=1}^{3N_{\text{atoms}}} \left( \frac{\partial v^{(l;\text{old})}}{\partial x_{j}} \right) \delta x_{j} \end{bmatrix} \quad (3.81)$$

$$+ \sum_{k=1}^{3N_{\text{atoms}}-6(5)} \frac{\partial U(\mathbf{v}^{\text{old}})}{\partial v^{(k)}} \begin{bmatrix} \sum_{j=1}^{3N_{\text{atoms}}} \left( \frac{\partial^{2} v^{(k;\text{old})}}{\partial x_{i} \partial x_{j}} \right) \delta x_{j} \end{bmatrix}$$

A second application of the chain rule for second derivatives, together with the secant condition in Cartesian coordinates, allows us to simplify the second term

$$\sum_{j=1}^{3N_{\text{atoms}}} \left( \frac{\partial^2 v^{(k;\text{old})}}{\partial x_i \partial x_j} \right) \delta x_j = \sum_{j=1}^{3N_{\text{atoms}}} \sum_{m=1}^{N_{\text{int}}} \left( \frac{\partial^2 v^{(k;\text{old})}}{\partial q_m \partial x_j} \left( \frac{\partial q_m^{\text{old}}}{\partial x_i} \right) + \frac{\partial v^{(k;\text{old})}}{\partial q_m} \frac{\partial^2 q_m^{\text{old}}}{\partial x_i \partial x_j} \right) \delta x_j$$
$$= \left[ \left( \mathbf{B}^{\text{old}} \right)^T \delta \mathbf{V} + \left( \delta \mathbf{B} \right)^T \mathbf{V}^{\text{old}} \right]_{ik}$$
(3.82)

The first term is simplified using the chain rule for first derivatives,

$$\frac{\partial v^{(k;\text{old})}}{\partial x_i} = \sum_{m=1}^{N_{\text{int}}} \frac{\partial v^{(k;\text{old})}}{\partial q_m} \frac{\partial q_m^{\text{old}}}{\partial x_i} = \left[ \left( \mathbf{B}^{\text{old}} \right)^T \mathbf{V}^{\text{old}} \right]_{ik}$$
(3.83)

and the secant condition. This gives the key intermediate result,

$$\delta \mathbf{g}_{x} \approx \left(\mathbf{B}^{\text{old}}\right)^{T} \mathbf{V}^{\text{old}} \mathbf{H}_{v}^{\text{old}} \delta \mathbf{v} + \left[\left(\mathbf{B}^{\text{old}}\right)^{T} \delta \mathbf{V} + \left(\delta \mathbf{B}\right)^{T} \mathbf{V}^{\text{old}}\right] \mathbf{g}_{v}^{\text{old}}$$

$$= \left(\mathbf{B}^{\text{old}}\right)^{T} \mathbf{V}^{\text{old}} \mathbf{H}_{v}^{\text{old}} \delta \mathbf{v} + \left(\mathbf{B}^{\text{old}}\right)^{T} \delta \mathbf{V} \mathbf{g}_{v}^{\text{old}} + \left(\delta \mathbf{B}\right)^{T} \mathbf{g}_{q}^{\text{old}}$$
(3.84)

Multiplying both sides of this equation by  $\mathbf{V}^T \mathbf{B}^+$  and rearranging terms, we get

$$\mathbf{H}_{\nu}^{\text{old}} \boldsymbol{\delta} \mathbf{v} \approx \boldsymbol{\delta} \mathbf{g}_{\nu} - \left(\mathbf{V}^{\text{old}}\right)^{T} \left(\mathbf{B}^{\text{old}}\right)^{+} \left(\left(\mathbf{B}^{\text{old}}\right)^{T} \boldsymbol{\delta} \mathbf{V} \mathbf{g}_{\nu}^{\text{old}} + \left(\boldsymbol{\delta} \mathbf{B}\right)^{T} \mathbf{g}_{q}^{\text{old}}\right).$$
(3.85)

This is the expression we use in our program; it approximately accounts for the dependence of the v-coordinates (and also the underlying q-coordinates) on the molecular geometry.

An alternative secant condition can be obtained starting from the expression,

$$\begin{bmatrix} \mathbf{H}_{\nu}^{\text{old}} \delta \mathbf{v} \end{bmatrix}_{i} = \sum_{j=1}^{3N_{\text{atoms}}-6(5)} \frac{\partial^{2} U(\mathbf{v}^{\text{old}})}{\partial \nu^{(i)} \partial \nu^{(j)}} \delta v^{(j)}$$

$$= \sum_{k=1}^{3N_{\text{atoms}}} \left( \frac{\partial x_{k}^{\text{old}}}{\partial \nu^{(i)}} \right) \left( \sum_{j=1}^{3N_{\text{atoms}}-6(5)} \frac{\partial^{2} U(\mathbf{x}^{\text{old}})}{\partial x_{k} \partial \nu^{(j)}} \delta v^{(j)} \right)$$

$$+ \sum_{k=1}^{3N_{\text{atoms}}} \frac{\partial U(\mathbf{x}^{\text{old}})}{\partial x_{k}} \left( \sum_{j=1}^{3N_{\text{atoms}}-6(5)} \frac{\partial^{2} x_{k}^{\text{old}}}{\partial \nu^{(i)} \partial \nu^{(j)}} \delta v^{(j)} \right)$$
(3.86)

Using the identity,

$$\sum_{j=1}^{3N_{\text{atoms}}-6(5)} \frac{\partial^2 x_k^{\text{old}}}{\partial v^{(i)} \partial v^{(j)}} \,\delta v^{(j)} = \sum_{m=1}^{N_{\text{int}}} \left( \frac{\partial q^{(m)}}{\partial v^{(i)}} \right)^{3N_{\text{atoms}}-6(5)} \frac{\partial^2 x_k^{\text{old}}}{\partial q^{(m)} \partial v^{(j)}} \,\delta v^{(j)} + \sum_{m=1}^{N_{\text{int}}} \frac{\partial x_k^{\text{old}}}{\partial q^{(m)}} \sum_{j=1}^{N_{\text{atoms}}-6(5)} \left( \frac{\partial^2 q^{(m)}}{\partial v^{(i)} \partial v^{(j)}} \right) \delta v^{(j)} = \left[ \left( \mathbf{V}^{\text{old}} \right)^T \left( \delta \mathbf{B}^+ \right)^T + \left( \delta \mathbf{V}^T \right) \left( \left( \mathbf{B}^{\text{old}} \right)^+ \right)^T \right]_{ik}$$
(3.87)

which gives the following secant condition,

$$\mathbf{H}_{\nu}^{\text{old}} \boldsymbol{\delta} \mathbf{v} = \boldsymbol{\delta} \mathbf{g}_{\nu} + \left( \left( \mathbf{V}^{\text{old}} \right)^{T} \left( \boldsymbol{\delta} \mathbf{B}^{+} \right)^{T} + \left( \boldsymbol{\delta} \mathbf{V}^{T} \right) \left( \left( \mathbf{B}^{\text{old}} \right)^{+} \right)^{T} \right) \mathbf{g}_{x}^{\text{old}}$$

$$= \boldsymbol{\delta} \mathbf{g}_{\nu} + \left( \mathbf{V}^{\text{old}} \right)^{T} \left( \boldsymbol{\delta} \mathbf{B}^{+} \right)^{T} \mathbf{g}_{x}^{\text{old}} + \left( \boldsymbol{\delta} \mathbf{V}^{T} \right) \mathbf{g}_{q}^{\text{old}}$$

$$(3.88)$$

Implicit in the assumptions of the quadratic model is that taking a step in the  $-\delta v$  direction from  $v^{new}$  is the same as taking a step in  $\delta v$  from  $v^{old}$ . Ergo

$$\mathbf{H}_{v}^{\text{new}} \boldsymbol{\delta} \mathbf{v} = \mathbf{H}_{v}^{\text{old}} \boldsymbol{\delta} \mathbf{v} \,. \tag{3.89}$$

This symmetry is not preserved by these secant updates. One can usually obtain a more accurate update by restoring the symmetry. For example, corresponding to Eq. (3.88), one has

$$\mathbf{H}_{\nu}\delta\mathbf{v} = \delta\mathbf{g}_{\nu} + \frac{1}{2} \left( \left( \mathbf{V}^{\text{old}} \right)^{T} \left( \delta\mathbf{B}^{+} \right)^{T} \mathbf{g}_{x}^{\text{old}} + \left( \mathbf{V}^{\text{new}} \right)^{T} \left( \delta\mathbf{B}^{+} \right)^{T} \mathbf{g}_{x}^{\text{new}} \right) + \frac{1}{2} \left( \delta\mathbf{V}^{T} \right) \left( \mathbf{g}_{q}^{\text{old}} + \mathbf{g}_{q}^{\text{new}} \right)$$
(3.90)

### Appendix 3.2. The set of reactions used for testing.

The following table lists the reactions we used for testing the transition-state finding methods studied in this chapter, along with the reduced coordinates for which the Hessian was approximated with the finite-difference method. Reactions 4, 10, 13, and 17 are taken from the test set of Grimme;<sup>92</sup> reaction 8 is taken from our own work on the epoxide hydrolase enzyme.<sup>37,93</sup> All the other reactions are taken from the Baker and Chan.<sup>42</sup>

Number	Reaction		Reduced coordinates
1	1,2-migration-(formyloxy) ethyl	$H = \begin{array}{c} H \\ H \\ H \\ 2 \\ 2 \\ 0 \\ H \end{array}$	R(1-2)
2	Ring opening bicycle[1.1.0] butane TS1	$ \begin{array}{c} \begin{array}{c} \begin{array}{c} \end{array}{}\\ \end{array}{}\\ \end{array}{}\\ \end{array}{}\\ \end{array}{}\\ \begin{array}{c} \end{array}{}\\ \end{array}{}\\ \end{array}} \begin{array}{c} \end{array}{}\\ \begin{array}{c} \end{array}{}\\ \begin{array}{c} \end{array}{}\\ \end{array}{}\\ \begin{array}{c} \end{array}{}\\ \end{array}{}\\ \begin{array}{c} \end{array}{}\\ \begin{array}{c} \end{array}{}\\ \end{array}{}\\ \end{array}{}\\ \begin{array}{c} \end{array}{}\\ \end{array}{}\\ \end{array}{}\\ \begin{array}{c} \end{array}{}\\ \end{array}{}\\ \begin{array}{c} \end{array}{}\\ \end{array}{}\\ \end{array}{}\\ \begin{array}{c} \end{array}{}\\ \end{array}{}\\ \end{array}{}\\ \begin{array}{c} \end{array}{}\\ \end{array}{}\\ \end{array}{}\\ \end{array}{}\\ \end{array}{}\\ \end{array}{}\\ \begin{array}{c} \end{array}{}\\ \end{array}{}\\ \end{array}{}\\ \end{array}{}\\ \end{array}{}\\ \end{array}{}\\ \end{array}{}\\ \end{array}{}$	R(1-2) R(1-3) ∡ 1-2-3

3	Ring opening bicycle[1.1.0] butane TS2	$ \begin{array}{cccccccccccccccccccccccccccccccccccc$	R(1-2) R(1-3)
4	$C_2H_6 + NH_2 \rightarrow C_2H_5 + NH_3$	$H \xrightarrow{H} H \xrightarrow{H} H \xrightarrow{H} H \xrightarrow{3} H \xrightarrow{H} $	R(1-2) R(2-3) R(1-3)
5	CH <sub>2</sub> CHOH → CH <sub>3</sub> CHO	$H_2C \xrightarrow{1}_{2 - H_3} \xrightarrow{1}_{H_2C} \xrightarrow{1}_{2 - H_2}$	R(1-3) R(2-3)
6	$CH_3O \rightarrow CH_2OH$	$H \xrightarrow{H} 1^{\circ} 0^{\circ} \xrightarrow{H} 1^{\circ} C \xrightarrow{0} H^{\circ} 2^{\circ} H^{\circ} H^{\circ} 2^{$	R(1-2) R(2-3)
7	cyclohexene →Butadiene + ethylene	$4 \bigoplus_{1}^{3} 2 \longrightarrow 1 = _{4} + 2 = 3$	R(1-2) R(3-4)
8	Epoxide hydrolase cluster model	$O = \begin{pmatrix} CH_3 \\ 3 \\ 3 \\ \end{pmatrix} + \begin{pmatrix} H_3C \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0$	R(1-2) R(1-3)

9	$H_2CNH \rightarrow HCNH_2$	$H \xrightarrow{3H}_{H} \xrightarrow{3H}_{H} \xrightarrow{3H}_{H} \xrightarrow{1}_{H} \xrightarrow{1}_{H}$	R(1-3) R(2-3)
10	$HF + CH_3 \rightarrow H+ FCH_3$	$\begin{array}{cccc} \cdot CH_3 + & F - H & \longrightarrow & H_3C - F & + & H \\ 1 & 2 & 3 & & & 1 & 2 & 3 \end{array}$	R(1-2) R(2-3)
11	$HCN \rightarrow HNC$	$N \equiv CH \longrightarrow H - N \equiv C^{-}$ $1 \ 2 \ 3 \qquad 3 \ 1 \ 2$	R(1-2) R(1-3)
12	$HCNH_2 \rightarrow HCN + H_2$	H H H H H H H H H H H H H H H H H H H	R(1-2) R(1-3)
13	hexatriene cyclozation	$\left  \begin{array}{c} \\ \\ \\ \\ \\ \\ 1 \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \\ \\ 1 \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \left  \begin{array}{c} \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \\ \\ \\ \\ \\ \\ \\ \end{array} \right  \xrightarrow{2} \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\$	R(1-2)
14	$2\text{HCN} + \text{N}_2 \rightarrow \text{s-tetrazine}$	N = N + N = H + N = H + N = H + N = H + 5 + 1 + 7 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1	$ \begin{array}{c} R(1-4) + R(1-7) + \\ R(3-6) \\ \measuredangle 3-4-5 \\ \measuredangle 6-7-8 \end{array} $
15	$SiH_3 + CH_2CH_3 \rightarrow SiH_2 + CH_3CH_3$	H H H H H H H H H H H H H H H H H H H	R(1-2)
16	CO + HCl →HCO + Cl	$\begin{array}{cccccccccccccccccccccccccccccccccccc$	R(1-2) R(2-3) ∡ 1-2-3
17	$OH + NH_3 \rightarrow H_2O + NH_2$	$H \rightarrow H + O \rightarrow H \rightarrow H^{0}_{3}_{2} + H^{1}_{1}_{1}_{1}_{1}_{1}_{1}_{1}_{1}_{1}_$	R(1-2) R(2-3)

Ph.D. Thesis –	Sandra Rabi;	<i>McMaster</i>	University -	Chemistry and	l Chemical B	Biology

18	$HNCCS \rightarrow HNC + CS$	$S = C = C = N$ $1  2  H \qquad \qquad$	R(1-2)
19	$H_2PO_4^- \rightarrow H_2O + PO_3^-$	$\begin{array}{c} 4 \\ 3 \\ 0 \\ - \\ 1 \\ 0 \\ - \\ - \\ - \\ - \\ + \\ - \\ - \\ - \\ - \\ -$	R(1-2) R(3-4)
20	$\begin{array}{c} CH_2CHCH_2\text{-}O\text{-}CHCH_2 \rightarrow \\ CH_2CHCH_2CH_2CHO \end{array}$	$\begin{array}{cccccccccccccccccccccccccccccccccccc$	R(1-2) R(3-4)

### 3.7 References

- (1) Schlegel, H. B. J. Comput. Chem. 2003, 24, 1514.
- (2) Schlegel, H. B. Wiley Interdisciplinary Reviews-Computational Molecular

Science 2011, 1, 790.

- (3) See Chapter 2.
- (4) Bakken, V.; Helgaker, T. J. Chem. Phys. 2002, 117, 9160.
- (5) Henkelman, G.; Uberuaga, B. P.; Jonsson, H. J. Chem. Phys. 2000, 113,

9901.

- (6) Henkelman, G.; Jonsson, H. J. Chem. Phys. 2000, 113, 9978.
- (7) Trygubenko, S. A.; Wales, D. J. J. Chem. Phys. 2004, 120, 2082.
- (8) Chu, J. W.; Trout, B. L.; Brooks, B. R. J. Chem. Phys. 2003, 119, 12708.
- (9) Maragakis, P.; Andreev, S. A.; Brumer, Y.; Reichman, D. R.; Kaxiras, E.

J. Chem. Phys. 2002, 117, 4651.

- (10) Galvan, I. F.; Field, M. J. J. Comput. Chem. 2008, 29, 139.
- (11) Sheppard, D.; Terrell, R.; Henkelman, G. J. Chem. Phys. 2008, 128,

134106.

- (12) Alfonso, D. R.; Jordan, K. D. J. Comput. Chem. 2003, 24, 990.
- (13) Gonzalez-Garcia, N.; Pu, J. Z.; Gonzalez-Lafont, A.; Lluch, J. M.; Truhlar,

D. G. J. Chem. Theory Comp. 2006, 2, 895.

(14) E, W. N.; Ren, W. Q.; Vanden-Eijnden, E. J. Chem. Phys. 2007, 126, 164103.

(15) E, W.; Ren, W. Q.; Vanden-Eijnden, E. Phys. Rev. B 2002, 66.

- (16) Ayala, P. Y.; Schlegel, H. B. J. Chem. Phys. 1997, 107, 375.
- (17) Burger, S. K.; Yang, W. J. Chem. Phys. 2007, 127, 164107.
- (18) Burger, S. K.; Yang, W. T. J. Chem. Phys. 2006, 124, 054109.
- (19) Koslover, E. F.; Wales, D. J. J. Chem. Phys. 2007, 127.
- (20) Ghasemi, S. A.; Goedecker, S. J. Chem. Phys. 2011, 135.
- (21) Granot, R.; Baer, R. J. Chem. Phys. 2008, 128.
- (22) Halgren, T. A.; Lipscomb, W. N. Chem. Phys. Lett. 1977, 49, 225.
- (23) Peng, C. Y.; Schlegel, H. B. Isr. J. Chem. 1993, 33, 449.
- (24) See Chapter 2.
- (25) Irikura, K. K.; Johnson, R. D. J. Phys. Chem. A 2000, 104, 2191.
- (26) Liu, Y. L.; Burger, S. K.; Dey, B. K.; Sarkar, U.; Janicki, M.; Ayers, P. W.

In Quantum Biochemistry; Matta, C. F., Ed.; Wiley-VCH: Boston, 2010.

- (27) Ohno, K.; Maeda, S. *Phys. Scr.* **2008**, *78*, 058122.
- (28) Maeda, S.; Watanabe, Y.; Ohno, K. Chem. Phys. Lett. 2005, 414, 265.
- (29) Maeda, S.; Ohno, K. J. Phys. Chem. A 2005, 109, 5742.
- (30) Ohno, K.; Maeda, S. Chem. Phys. Lett. 2004, 384, 277.
- (31) Liu, Y.; Ayers, P. W. J. Math. Chem. 2011, 49, 1291.
- (32) Burger, S. K.; Ayers, P. W. J. Chem. Theory Comp. 2010, 6, 1490.
- (33) Burger, S. K.; Liu, Y. L.; Sarkar, U.; Ayers, P. W. J. Chem. Phys. 2009,

130, 024103.

- (34) Dey, B. K.; Janicki, M. R.; Ayers, P. W. J. Chem. Phys. 2004, 121, 6667.
- (35) Dey, B. K.; Ayers, P. W. Mol. Phys. 2006, 104, 541.

- (36) See Chapter 2.
- (37) Burger, S. K.; Ayers, P. W. J. Chem. Phys. 2010, 132, 234110.
- (38) Heyden, A.; Bell, A. T.; Keil, F. J. J. Chem. Phys. 2005, 123.
- (39) Henkelman, G.; Jonsson, H. J. Chem. Phys. 1999, 111, 7010.
- (40) Schlegel, H. B. J. Comput. Chem. 1982, 3, 214.
- (41) Frisch, M. J.; Trucks, G. W.; Schlegel, H. B.; Scuseria, G. E.; Robb, M.

A.; Cheeseman, J. R.; Scalmani, G.; Barone, V.; Mennucci, B.; Petersson, G. A.;

Nakatsuji, H.; Caricato, M.; Li, X.; Hratchian, H. P.; Iszmaylov, A. F.; Bloino, J.; Zheng,

G.; Sonnenberg, J. L.; Hada, M.; Ehara, M.; Toyota, K.; Fukuda, R.; Hasegawa, J.; Ishida,

- M.; Nakajima, T.; Honda, Y.; Kitao, O.; Nakai, H.; Vreven, T.; Montgomery Jr., J. A.;
- Peralta, J. E.; Ogliaro, F.; Bearpark, M.; Heyd, J. J.; Brothers, E.; Kudin, K. N.;
- Staroverov, V. N.; Kobayashi, R.; Normand, J.; Raghavachari, K.; Rendell, A.; Burant, J.
- C.; Iyengar, S. S.; Tomasi, J.; Cossi, M.; Rega, N.; Millam, J. M.; Klene, M.; Knox, J. E.;
- Cross, J. B.; Bakken, V.; Adamo, C.; Jaramillo, J.; Gomperts, R.; Stratmann, R. E.;
- Yazyev, O.; Austin, A. J.; Cammi, R.; Pomelli, C.; Ochterski, J. W.; Martin, R. L.;
- Morokuma, K.; Zakrzewski, V. G.; Voth, G. A.; Salvador, P.; Dannenberg, J. J.;
- Dapprich, S.; Daniels, A. D.; Farkas, O.; Foresman, J. B.; Ortiz, J. V.; Cioslowski, J.;

Fox, D. J.; Gaussian Inc.: Wallingford CT, 2009.

- (42) Baker, J.; Chan, F. R. J. Comput. Chem. 1996, 17, 888.
- (43) See Chapter 2.
- (44) Baker, J.; Hehre, W. J. J. Comput. Chem. 1991, 12, 606.
- (45) Schlegel, H. B. Int. J. Quantum Chem. 1992, 243.

(46) Gordon, M. S.; Pople, J. A. J. Chem. Phys. 1968, 49, 4643.

(47) Pulay, P.; Fogarasi, G.; Pang, F.; Boggs, J. E. J. Am. Chem. Soc. 1979, 101, 2550.

(48) Fogarasi, G.; Zhou, X. F.; Taylor, P. W.; Pulay, P. J. Am. Chem. Soc.1992, 114, 8191.

- (49) Pulay, P.; Fogarasi, G. J. Chem. Phys. 1992, 96, 2856.
- (50) Baker, J.; Kessi, A.; Delley, B. J. Chem. Phys. 1996, 105, 192.
- (51) Baker, J.; Pulay, P. J. Chem. Phys. 1996, 105, 11100.
- (52) Baker, J.; Kinghorn, D.; Pulay, P. J. Chem. Phys. 1999, 110, 4986.
- (53) Baker, J.; Pulay, P. J. Comput. Chem. 2000, 21, 69.
- (54) von Arnim, M.; Ahlrichs, R. J. Chem. Phys. 1999, 111, 9183.
- (55) Peng, C. Y.; Ayala, P. Y.; Schlegel, H. B.; Frisch, M. J. J. Comput. Chem.

1996, 17, 49.

- (56) Farkas, O.; Schlegel, H. B. J. Mol. Struct.: THEOCHEM 2003, 666, 31.
- (57) Farkas, O.; Schlegel, H. B. J. Chem. Phys. 1998, 109, 7100.
- (58) Eckert, F.; Pulay, P.; Werner, H. J. J. Comput. Chem. 1997, 18, 1473.
- (59) Billeter, S. R.; Turner, A. J.; Thiel, W. PCCP 2000, 2, 2177.
- (60) Paizs, B.; Baker, J.; Suhai, S.; Pulay, P. J. Chem. Phys. 2000, 113, 6566.
- (61) Kudin, K. N.; Scuseria, G. E.; Schlegel, H. B. J. Chem. Phys. 2001, 114,

2919.

- (62) Bucko, T.; Hafner, J.; Angyan, J. G. J. Chem. Phys. 2005, 122.
- (63) Doll, K.; Dovesi, R.; Orlando, R. Theor. Chem. Acc. 2006, 115, 354.

- (64) Swart, M.; Bickelhaupt, F. M. Int. J. Quantum Chem. 2006, 106, 2536.
- (65) Scharfenberg, P. Chem. Phys. Lett. 1981, 79, 115.
- (66) Bofill, J. M.; Anglada, J. M. Theor. Chem. Acc. 2001, 105, 463.
- (67) Anglada, J. M.; Besalu, E.; Bofill, J. M.; Crehuet, R. J. Comput. Chem.

**2001**, *22*, 387.

- (68) Burger, S. K.; Liu, Y.; Sarkar, U.; Ayers, P. W. J. Chem. Phys. 2009, 130.
- (69) Gill, P. E.; Murray, W.; Wright, M. H. *Practical optimization*; Academic Press: London ; Toronto, 1981.
- (70) Dennis, J. E.; Schnabel, R. B. *Numerical methods for unconstrained optimization and nonlinear equations*; Prentice-Hall: Englewood Cliffs, N.J., 1983.
  - - (71) Powell, M. J. D. *Mathematical Programming* **1971**, *1*, 26.
    - (72) Broyden, C. G. Journal of the Institute of Mathematics and Its

Applications 1970, 6, 76.

- (73) Fletcher, R. Computer Journal **1970**, *13*, 317.
- (74) Goldfarb, D. *Mathematics of Computation* **1970**, *24*, 23.
- (75) Shanno, D. F. *Mathematics of Computation* **1970**, *24*, 647.
- (76) Shanno, D. F.; Kettler, P. C. Mathematics of Computation 1970, 24, 657.
- (77) Bofill, J. M. J. Comput. Chem. 1994, 15, 1.
- (78) Conn, A. R.; Gould, N. I. M.; Toint, P. L. Mathematical Programming

**1991**, *50*, 177.

(79) Powell, M. J. D. In *Nonlinear Programming 3*; Mangasarian, O., Meyer,M., Robinson, S., Eds.; Academic Press: New York, 1978, p 27.

(80) Nocedal, J.; Wright, S. J. *Numerical Optimization*; Springer-Verlag: New York, 1999.

(81) Cerjan, C. J.; Miller, W. H. J. Chem. Phys. 1981, 75, 2800.

(82) Simons, J.; Jorgensen, P.; Taylor, H.; Ozment, J. J. Phys. Chem. 1983, 87,

2745.

(83) Nichols, J.; Taylor, H.; Schmidt, P.; Simons, J. J. Chem. Phys. 1990, 92,

340.

(84) Simons, J.; Nichols, J. Int. J. Quantum Chem. 1990, 263.

(85) Helgaker, T. Chem. Phys. Lett. 1991, 182, 503.

(86) Culot, P.; Dive, G.; Nguyen, V. H.; Ghuysen, J. M. Theor. Chim. Act.

**1992**, *82*, 189.

(87) Press, W. H.; Teukolsky, S. A.; Vetterling, W. T.; Flannery, B. P.;

Cambridge UP: New York, 1992; Vol. 2nd.

(88) Banerjee, A.; Adams, N.; Simons, J.; Shepard, R. J. Phys. Chem. 1985, 89,

52.

- (89) Baker, J. J. Comput. Chem. 1986, 7, 385.
- (90) Besalu, E.; Bofill, J. M. Theor. Chem. Acc. 1998, 100, 265.
- (91) Anglada, J. M.; Bofill, J. M. Int. J. Quantum Chem. 1997, 62, 153.
- (92) Goerigk, L.; Grimme, S. J. Chem. Theory Comp. 2010, 6, 107.
- (93) S. Rabi, S. K. Burger, and P. W. Ayers (in preparation).

## **Chapter 4**

## Systematic Assessment of Transition-State Optimization Methods Using Random Transition-State Guesses for a Database of 131 Reactions

### 4.1 Statement of the Problem

We recently presented in chapter 3 a new transition-state optimization algorithm in which the portion of the Hessian associated with a few reactive coordinates was accurately approximated using finite differences, while the remainder of the Hessian was updated using quasi-Newton approaches. In this work, we develop a protocol for testing transition-state optimizers and compare our new method to conventional approaches, specifically the Berny optimizer and QST3 methods, as implemented in *Gaussian*. Our testing protocol is based on a database of 131 diverse chemical reactions, superseding many of the standard reaction sets. We then add random vectors of specified magnitude to the exact transition state structures of these reactions and ascertain whether a method can recover the initial transition-state structure. Our new method is marginally slower than conventional approaches, but it converges more frequently when the initial guess for the transition state is poor.

### 4.2 Introduction

With the proliferation of algorithms for optimizing wavefunctions and molecular geometries, the need for systematic approaches to compare methods has become acute. Most often, people test new approaches against a small dataset, causing skeptical readers to wonder if the dataset was designed to demonstrate the advantages, and avoid the

disadvantages, of their new method. This paper presents a protocol for assessing the quality of numerical optimizations of transition-states, but the same general principles can be used in other contexts also.

The ingredients of the proposed protocol are presented in section 4.3: (A) a broad database of chemically-relevant test cases, (B) a systematic procedure for generating initial guesses of deteriorating quality, and (C) a method for comparing different methods' performance across the database.

We are motivated by our recent work to develop a more robust transition-state finding algorithm, as well as the proliferation of methods for locating transition states on potential energy surfaces.<sup>1-3</sup> Our goal is to assess traditional transition-state optimizers, that is, methods that start from a (hopefully good) initial guess for the transition state and then optimize the transition state using some sort of quasi-Newton method.<sup>4-13</sup> We only consider approaches using redundant internal coordinates<sup>14</sup> because we believe that methods based on Cartesian or internal (z-matrix) coordinates are uncompetitive.<sup>15-19</sup> Neither will we consider methods based on repeated computation of the exact Hessian of the molecular potential energy surface, as they tend to be prohibitively expensive for large molecules. Finally, we do not consider single-ended,<sup>5,20-34</sup> double-ended,<sup>35-42</sup> and surface-walking approaches.<sup>36,43-56</sup> These methods tend to be more expensive than straightforward optimization, and are most efficiently employed for finding a good initial guess for a conventional transition-state optimizer.

We compare three transition-state optimizers with different degrees of sophistication. The most traditional method is the Berny optimizer,<sup>4,17</sup> which is the default

177

method in the *Gaussian* program.<sup>57</sup> The default Berny optimizer for transition-states uses a trust radius and the Bofill approximation to the quasi-Newton Hessian.<sup>11</sup> The step is computed using the rational-function optimization approach to eigenvector following.<sup>5,32</sup>

The method we recently presented, called *Saddle*, is slightly more sophisticated because key coordinates involved in the chemical reaction are identified, and the Hessian elements that involve differentiation with respect to one or more of these coordinates are approximated with finite-differences, rather than through the quasi-Newton Hessian.<sup>12,58</sup> This is the only philosophical difference between *Saddle* and the Berny optimizer, but our method also features a more robust choice for the internal coordinates,<sup>59</sup> a different method for converting from internal to Cartesian coordinates,<sup>60</sup> and a different trust-radius method.<sup>61</sup>

Of greatest sophistication is the QST3 method,<sup>62</sup> which requires not only an initial guess for the transition-state geometry, but also the structures of the reactants and products. The QST3 algorithm performs an initial refinement of the molecular geometry, after which the traditional Berny algorithm is invoked.

In subsequent sections, we describe our testing protocol, compare the performance of Berny optimizer, our optimizer, and the QST3 method, and state some conclusions.

### 4.3 Testing Protocol

### A. A Database of Chemical Reactions for Testing Transition-State Optimizers

In order to test a computational algorithm, one first needs to construct a broad and relevant database. The present database is based on our previous work.<sup>63</sup> but we added one additional transition-metal reaction and eliminated one atom-exchange reaction. Most of the reactions were taken from reaction sets that were used for testing density functionals.<sup>64-69</sup> The Baker-Chan reactions, which are quite commonly used for testing transition-state optimizers, are also included.<sup>19</sup> Several reactions were taken from the meticulous work on potential-energy curves performed by the Toro-Labbe group,<sup>70-72</sup> the topological reactivity studies from the Jenkins group,<sup>73</sup> the chemical vapour deposition reactions of Coyle *et al.*,<sup>74</sup> and (often unpublished) research from our group.<sup>12,75,76</sup> Thirty reactions were constructed by adding functional groups to molecules from the other reactions; this produced several sterically-hindered reactions and requires transition-state optimizers to distinguish between true reactions and low-barrier conformational changes. All transition states were determined using HF/6-31++G(d,p) except for the coppercontaining reactions of Coyle et al.,<sup>74</sup> which were determined using B3LYP/6-31G(d). We used the *Gaussian* program.<sup>57</sup>

#### **B.** A Systematic Method for Generating Initial Guesses of Decreasing Quality

We will assess the computational robustness and efficiency of these transitionstate optimizers by providing initial guesses of varying quality. Our approach is to construct initial guesses that are random perturbations of the true transition state. We start by generating a random vector,  $\mathbf{a}$ , in Cartesian coordinates. This vector is projected using the Wilson **B** matrix and its generalized inverse, then normalized,

$$\hat{\mathbf{u}} = \frac{\mathbf{B}^{+}\mathbf{B}\mathbf{a}}{\left|\mathbf{B}^{+}\mathbf{B}\mathbf{a}\right|} \tag{4.1}$$

This specifies a molecular frame by removing ambiguities due to the center-of-mass and overall molecular orientation and produces a random perturbation of the internal degrees of freedom, with unit length. The Cartesian coordinates of the molecule are then adjusted by making a random move away from the true transition-state geometry,

$$\mathbf{x} = \mathbf{x}_{\text{t.s.}} + \hat{\mathbf{u}} \mathcal{E} \sqrt{3N_{\text{atoms}}}$$
(4.2)

The prefactor is designed to ensure that the average change in a specific Cartesian coordinate is the same in few-atom and many-atom molecules. We generated 10 random initial guesses for each value of  $\varepsilon$ , increasing  $\varepsilon$  until the rate of convergence for the methods we tested deteriorated dramatically. Specifically, we chose  $\varepsilon = 0.05$ , 0.1, 0.2, and 0.3 Bohr.

We observed that the initial geometries obtained from Eq. (4.2) were often not especially realistic, largely because portions of the molecule (e.g., atoms and functional groups that are spectators to the chemical reaction in question) are deformed away from equilibrium, which is rarely true in reasonable initial guesses. This motivated us to explore another protocol, in which only the key chemical coordinates used by *Saddle* were allowed to deform. To do this, we generated a random vector in redundant internal coordinates with nonzero components for the reduced-dimensional set of key chemical coordinates only. We projected this vector into the space of redundant internal coordinates,

$$\mathbf{v} = \mathbf{B}\mathbf{B}^{+}\mathbf{a}_{\text{reduced}} \tag{4.3}$$

and then generated the change in the Cartesian coordinates using the method from chapter 2. That is, we minimized,

$$\min_{\mathbf{x}_{\kappa}} \left| \mathbf{q}(\mathbf{x}_{\kappa}) - (\mathbf{q}_{\text{t.s.}} + \kappa \mathbf{v}) \right|^2, \qquad (4.4)$$

where  $\mathbf{q}_{\text{t.s.}}$  is the redundant internal coordinates of the transition-state and  $\kappa$  is chosen so that the distance of the perturbed Cartesian coordinates from the Cartesian coordinates of the true transition state is  $\mathcal{E}\sqrt{N_{\text{reduced}}}$ , where  $N_{\text{reduced}}$  is the number of redundant internal coordinates that are perturbed. In order to remove rotational and translational motion, the Cartesian distance is computed using the singular-value-decomposition version of Kabsch's alignment algorithm from chapter 2.<sup>77</sup> Once again we generated 10 random initial guesses for each choice of  $\varepsilon$ . Since the initial geometries obtained by this procedure were more accurate, calculations starting from more severely perturbed geometries still converged well, and we chose a larger value of  $\varepsilon$  ( $\varepsilon = 0.4$  Bohr) for our final value.

### C. A Criterion for Determining Whether a Method Converges to the Correct Transition State

Because some of the reactions have multiple chemically-equivalent transition states (e.g., transition states that differ only due to symmetry, or conformational differences of very distant functional groups), we need to develop a criterion for whether a calculation converges to the correct transition state.

To establish whether an approximate transition state is chemically equivalent to the true transition state, one should consider both the energy and the geometry of the transition-state. Since conformation changes (e.g., methyl rotations) far from the region of the reagents where the molecule occurs should not affect whether we accept the transition state, and since chemically-equivalent atoms are interchangeable, we use a criterion based on interatomic distances. Specifically, we identify as *quasi-reduced atoms* all atoms that are (a) either used to define the reduced coordinates or (b) bonded to a "reduced atom" by a regular (not interfragment or auxiliary/Urey-Bradley) bond. We then compute all the interatomic distances between the quasi-reduced atoms and sort the lists of interatomic distances in increasing order. Next, we compute the percent deviation between respective distances in the list. I.e., for the  $k^{th}$ -smallest interatomic distances in the list, we compute

$$p^{(k)} = \frac{\left| R_{ij}^{(k;\text{computed})} - R_{ij}^{(k;\text{exact})} \right|}{\frac{1}{2} \left( R_{ij}^{(k;\text{computed})} + R_{ij}^{(k;\text{exact})} \right)}$$
(4.5)

We would like to force small deviations (ca. 2%) on interatomic distances corresponding to chemical bonds while allowing larger deviations (ca. 5%) on interatomic distances corresponding to 1-3 or 1-4 interactions. This is achieved by defining the measure

$$q^{(k)} = \exp\left[-\left(\frac{\frac{1}{2}\left(R_{ij}^{(k;\text{computed})} + R_{ij}^{(k;\text{exact})}\right)}{4}\right)^2\right]p^{(k)}$$
(4.6)

If the energy of the located transition state is more than .001 a.u. from the correct energy or  $q^{(k)} \ge .02$  for any interatomic distances between quasi-reduced atoms, then we consider the calculation to have converged to the wrong transition state; this is treated as a convergence failure. We found that our conclusions are not sensitive to moderate changes in the energy criterion or the threshold for the geometric criterion.

### 4.4 Results: Comparing the Performance of Transition-State Optimizers

We assess the relative performance of the standard Berny optimizer, our method (*Saddle*), and the QST3 method by considering the number of gradient calculations required by each method when it converges to the correct transition state, averaged over all reactions and all initial guesses for which the method converged. Since the computational cost is dominated by the computation of the gradient, the relative number of gradient calculations represents the relative cost of the methods we consider. We also consider the fraction of the initial guesses that converge.

In Table 4.1, we compare the number of iterations required to converge to the correct transition state when all the atoms are perturbed, as described in Eq. (4.2). Unsurprisingly, the number of iterations increases dramatically as the quality of the initial guess deteriorates. For good initial guesses, our method requires slightly fewer iterations,

on average, than the Berny optimizer and QST3. In addition, QST3 is slower than the Berny optimizer; this is because the pre-processing stage in the QST3 method is unnecessary for good initial guesses. As the quality of the initial guess deteriorates, QST3 becomes better than the Berny optimizer and, for our worst initial guesses ( $\epsilon = .3$  Bohr), QST3 converges slightly faster than our new method. These results, however, are somewhat misleading: one reason that QST3 is slower than the Berny optimizer is that it converges more reactions, and the difficult reactions that converge with QST3, but not the Berny optimizer, tend to require more iterations. Similarly, our method is slower for bad initial guesses ( $\epsilon = .3$  Bohr) mainly because it converges more reactions (83.6%, vs. 77.6% with QST3).

While the differences in the computational costs of the methods are practically insignificant (within 25%), the differences in the rate of convergence are striking. (See Table 4.2.) All methods converge very well for good ( $\varepsilon = 0.1$  Bohr) and excellent ( $\varepsilon = 0.05$  Bohr) initial guesses, with the Berny optimizer actually outperforming QST3 for these easy cases. (It seems that the pre-processing state of QST3 sometimes leads the method away from the initial transition state when the initial guess is excellent.) For poor initial guesses, QST3 is decisively better than the Berny optimizer and our method converges about 8% more reactions than QST3. For the worst initial guesses ( $\varepsilon = 0.3$  Bohr), only our method converges to the targeted transition state more than 75% of the time.

Further insight into the relative performance of the methods for poor initial guesses ( $\varepsilon = 0.3$  Bohr) can be gleaned from a performance plot of the percent of reactions

184

that have converged in a set number of iterations; see Figure 4.1. This plot reveals that QST3 is very efficient when it converges, but for reactions that require more than about 40 steps of the QST3 optimizer, or 20 steps of the Berny optimizer, our method is faster. This is consistent with our previous observations. The strength of our method is its robustness; other methods tend to work better in the "easy" cases, but when other methods struggle to converge, the extra information about the potential energy surface in the vicinity of the transition state that our method incorporates is useful. In addition, even when our new method is slower than existing approaches, it is not uncompetitive.

**Table 4.1.** The number of steps, averaged over all transition-state guesses of a given quality for all 131 reactions, required for various methods to converge to the targeted transition state. The initial guesses are constructed by randomly perturbing all the Cartesian coordinates of the correct transition-state structure by a specified amount,  $\varepsilon$ , as expressed in Eqs. (4.1) and (4.2). If, for a given initial guess, a computational method fails (crashes), does not converge within 200 iterations, or converges to the wrong transition state, then that calculation is not included in the average.

ε (Bohr)	0.05	0.1	0.2	0.3
Method				
Our Method (Saddle)	8.9	17.4	35.3	56.0
Gaussian (standard Berny)	12.6	24.2	39.2	47.5
Gaussian (QST3)	15.7	27.3	42.9	55.8

**Table 4.2.** The percentage of transition-state guesses of a given quality that converge to the targeted transition state for a given method. The initial guesses are constructed by randomly perturbing all the Cartesian coordinates of the correct transition-state structure by a specified amount,  $\varepsilon$ , as expressed in Eqs. (4.1) and (4.2). Methods are considered to fail if the program crashes, fails to converge within 200 iterations, or converges to the wrong transition state.

ε (Bohr)	0.05	0.1	0.2	0.3
Method				
Our Method (Saddle)	99.8%	99.5%	93.1%	77.1%
Gaussian (standard Berny)	96.1%	87.8%	59.5%	32.3%
Gaussian (QST3)	93.9%	89.0%	80.8%	69.0%

**Figure 4.1.** A performance plot indicating the fraction of calculations, using different methods, that have converged to the targeted transition state within a given number of gradient calculations. The initial guesses are constructed by randomly perturbing all the Cartesian coordinates of the correct transition-state structure by  $\varepsilon = 0.3$  Bohr. Methods are considered to fail if the program crashes, fails to converge within 200 iterations, or converges to



the wrong transition state.

As mentioned in section 2.3.B, the initial geometries obtained by deforming all the Cartesian coordinates are unrealistic because portions of the molecule for which it is easy to make a good initial guess are dramatically deformed. In general, it is only the portions of the molecule that are directly involved in the chemical reaction for which it is difficult to construct a guess geometry. This led us to the second method for constructing initial guesses, in which only the "reduced" coordinates involved in the chemical reaction are altered. Results for this method of generating initial guesses are reported in Tables 4.3 and 4.4, and Figure 4.2.

With these more reasonable initial guesses, our conclusions change somewhat. The standard Berny algorithm still converges more rapidly than QST3 for good initial guesses; this is mostly due because the QST3 procedure for refining the initial guess is counterproductive when the initial guess is already excellent. The average number of steps required by our method is typically somewhat larger than both the Berny and QST3 methods, though it is not uncompetitive. For the worst initial guesses, QST3 is slightly faster than the Berny algorithm, which is faster than our approach. However, all of the methods are competitive in terms of computational speed.

Where our method, and also QST3, has a decisive advantage is in terms of robustness. Both QST3 and our method converge significantly more often for less-thanexcellent initial guesses ( $\varepsilon = 0.2$  and 0.4 Bohr). Only our method converges more than 85% of the reactions for the worst set of initial guesses.

The performance plot in Figure 4.2 reveals additional details about the performance of the methods for poor initial guesses. The Berny optimizer is best for the easy cases, which converge in fifteen or fewer iterations; this reflects the costly pre-processing stages of QST3 and our method (in *Saddle*, most of the finite-difference calculations are performed in the first few iterations). For cases that are somewhat more

difficult, the QST3 method performs very well. It is mainly for the most intractable cases, which require 50 or more iterations to converge, where our method distinguishes itself. Our method is competitive, however, because it requires, on average, less than 5 extra iterations.

We also considered a less stringent criterion for transition-state convergence. We removed the restriction  $(\max(q^{(k)}) < 0.02)$  on the error in the transition-state structure and relaxed the energetic criterion, so that the method was judged to have correctly located the transition state if the energy of the structure it located was within .002 a.u. of the correct value. Relaxing the transition-state-acceptance criterion slightly improved the performance of conventional approaches, but had negligible effects on *Saddle*'s performance. (Specifically, the percentage of *Berny* and QST3 calculations that were judged to have converged increased by at most 3%, and the average number of iterations decreased by at most 2 iterations.) This suggests that our method is slightly less prone to converging to incorrect transition states. This is not surprising because, alone among these methods, *Saddle* has information about which are the key chemical coordinates that characterize the transition-state of interest.
**Table 4.3.** The number of steps, averaged over all transition-state guesses of a given quality for all 131 reactions, required for various methods to converge to the targeted transition state. The initial guesses are constructed by randomly perturbing the key chemical coordinates of correct transition-state structure so that the deformed structure differs from the true transition-state structure by a specified amount, ε. (Cf. Eqs. (4.3) and (4.4).) If, for a given initial guess, a computational method fails (crashes), does not converge within 200 iterations, or converges to the wrong transition state, then that calculation is not included in the average.

ε (Bohr)	0.05	0.1	0.2	0.4
Method				
Our Method (Saddle)	9.5	11.4	14.2	26.3
Gaussian (standard Berny)	3.8	5.7	9.4	15.3
Gaussian (QST3)	7.5	8.4	12.4	18.2

**Table 4.4.** The percentage of transition-state guesses of a given quality that converge to the targeted transition state for a given method. The initial guesses are constructed by randomly perturbing the key chemical coordinates of correct transition-state structure so that the deformed structure differs from the true transition-state structure by a specified amount, ε. Methods are considered to fail if the program crashes, fails to converge within 200 iterations, or converges to the wrong transition state.

ε (Bohr)	0.05	0.1	0.2	0.4
Method				
Our Method (Saddle)	99.7%	99.9%	98.0%	89.2%
Gaussian (standard Berny)	99.5%	99.1%	90.0%	59.2%
Gaussian (QST3)	96.0%	94.8%	90.4%	79.0%

**Figure 4.2.** A performance plot indicating the fraction of calculations, using different methods, that have converged to the targeted transition state within a given number of gradient calculations. The initial guesses are constructed by randomly perturbing the key chemical coordinates of the correct transition-state structure so that the deformed structure differs from the true transition-state structure by a specified amount,  $\varepsilon = 0.4$  Bohr. Methods are considered to fail if the program crashes, fails to converge within 200 iterations, or converges to the wrong transition state.



### 4.5 Summary

The goals of this chapter are to present a testing protocol for transition-state optimization methods and to compare our recent approach to the popular methods from the *Gaussian* program. The first key ingredient in our testing protocol is a large and broad database of 131 chemical reactions; all of the 130 reactions provided in chapter 2, with the addition of the following reaction:



Next, we designed a method for generating initial guesses of decreasing quality; this was done by making random perturbations, of specified type and magnitude, away from the optimized transition-state geometry. Finally, we assessed how well different methods, specifically (1) the standard Berny optimization algorithm for Gaussian, (2) the method we recently presented, which uses finite differences to approximate key components of the molecular Hessian, and (3) the QST3 method from Gaussian, which uses information about the reactant and product geometry to construct better initial guesses for the transition state optimizer. We can recommend using the QST3 method and our method. The QST3 method is slightly faster (by about five iterations), while our method is significantly more robust. It would be interesting to see if we could further improve our method's performance by using information about the reactant and product geometries, in the spirit of the QST3 method.

### 4.6 **References**

(1) Schlegel, H. B. J. Comput. Chem. 2003, 24, 1514.

(2) Schlegel, H. B. Wiley Interdisciplinary Reviews-Computational Molecular Science **2011**, *1*, 790.

(3) Wales, D. J. *Energy landscapes*; Cambridge University Press: Cambridge,UK ; New York, 2003.

- (4) Schlegel, H. B. J. Comput. Chem. **1982**, *3*, 214.
- (5) Baker, J. J. Comput. Chem. **1986**, 7, 385.
- (6) Bofill, J. M. Int. J. Quantum Chem. 2003, 94, 324.
- (7) Besalu, E.; Bofill, J. M. *Theor. Chem. Acc.* **1998**, *100*, 265.
- (8) Anglada, J. M.; Bofill, J. M. J. Comput. Chem. 1998, 19, 349.
- (9) Anglada, J. M.; Bofill, J. M. Int. J. Quantum Chem. 1997, 62, 153.
- (10) Bofill, J. M. Chem. Phys. Lett. 1996, 260, 359.
- (11) Bofill, J. M. J. Comput. Chem. 1994, 15, 1.
- (12) Burger, S. K.; Ayers, P. W. J. Chem. Phys. 2010, 132, 234110.
- (13) Henkelman, G.; Jonsson, H. J. Chem. Phys. 1999, 111, 7010.
- (14) Pulay, P.; Fogarasi, G.; Pang, F.; Boggs, J. E. J. Am. Chem. Soc. 1979,

101, 2550.

- (15) Baker, J.; Hehre, W. J. J. Comput. Chem. 1991, 12, 606.
- (16) Baker, J. J. Comput. Chem. **1993**, 14, 1085.
- (17) Peng, C. Y.; Ayala, P. Y.; Schlegel, H. B.; Frisch, M. J. J. Comput. Chem.

**1996**, *17*, 49.

- (18) Schlegel, H. B. Int. J. Quantum Chem. 1992, 243.
- (19) Baker, J.; Chan, F. R. J. Comput. Chem. **1996**, 17, 888.
- (20) Burkert, U.; Allinger, N. L. J. Comput. Chem. 1982, 3, 40.
- (21) Williams, I. H.; Maggiora, G. M. Theochem-Journal of Molecular

Structure 1982, 6, 365.

- (22) Scharfenberg, P. Chem. Phys. Lett. 1981, 79, 115.
- (23) Rothman, M. J.; Lohr, L. L. Chem. Phys. Lett. 1980, 70, 405.
- (24) Hirsch, M.; Quapp, W. J. Comput. Chem. 2002, 23, 887.
- (25) Quapp, W.; Hirsch, M.; Imig, O.; Heidrich, D. J. Comput. Chem. 1998, 19,

1087.

(26) Crehuet, R.; Bofill, J. M.; Anglada, J. M. Theor. Chem. Acc. 2002, 107,

130.

(27) Peters, B.; Heyden, A.; Bell, A. T.; Chakraborty, A. J. Chem. Phys. 2004,

120, 7877.

- (28) Quapp, W. J. Chem. Phys. 2005, 122, 174106.
- (29) Bofill, J. M.; Quapp, W.; Caballero, M. J. Chem. Theory Comp. 2012, 8,

927.

- (30) Bofill, J. M.; Quapp, W. J. Chem. Phys. 2011, 134, 074101.
- (31) Cerjan, C. J.; Miller, W. H. J. Chem. Phys. 1981, 75, 2800.
- (32) Banerjee, A.; Adams, N.; Simons, J.; Shepard, R. J. Phys. Chem. 1985, 89,

52.

(33) Simons, J.; Jorgensen, P.; Taylor, H.; Ozment, J. J. Phys. Chem. 1983, 87,2745.

(34) Culot, P.; Dive, G.; Nguyen, V. H.; Ghuysen, J. M. *Theor. Chim. Act.***1992**, 82, 189.

(35) Carr, J. M.; Trygubenko, S. A.; Wales, D. J. J. Chem. Phys. 2005, 122,
234903.

- (36) Trygubenko, S. A.; Wales, D. J. J. Chem. Phys. 2004, 120, 2082.
- (37) Quapp, W.; Bofill, J. M. J. Comput. Chem. 2010, 31, 2526.
- (38) Koslover, E. F.; Wales, D. J. J. Chem. Phys. 2007, 127.
- (39) E, W.; Ren, W. Q.; Vanden-Eijnden, E. Phys. Rev. B 2002, 66.
- (40) Ayala, P. Y.; Schlegel, H. B. J. Chem. Phys. 1997, 107, 375.
- (41) Burger, S. K.; Yang, W. J. Chem. Phys. 2007, 127, 164107.
- (42) Burger, S. K.; Yang, W. T. J. Chem. Phys. 2006, 124, 054109.
- (43) Irikura, K. K.; Johnson, R. D. J. Phys. Chem. A 2000, 104, 2191.
- (44) Liu, Y. L.; Burger, S. K.; Dey, B. K.; Sarkar, U.; Janicki, M.; Ayers, P. W.

In Quantum Biochemistry; Matta, C. F., Ed.; Wiley-VCH: Boston, 2010.

- (45) Ohno, K.; Maeda, S. *Phys. Scr.* **2008**, *78*, 058122.
- (46) Ohno, K.; Maeda, S. Chem. Phys. Lett. 2004, 384, 277.
- (47) Dey, B. K.; Ayers, P. W. Mol. Phys. 2006, 104, 541.
- (48) Dey, B. K.; Janicki, M. R.; Ayers, P. W. J. Chem. Phys. 2004, 121, 6667.
- (49) Liu, Y. L.; Burger, S. K.; Ayers, P. W. J. Math. Chem. 2011, 49, 1915.
- (50) Burger, S. K.; Ayers, P. W. J. Chem. Theory Comp. 2010, 6, 1490.

(51) Henkelman, G.; Uberuaga, B. P.; Jonsson, H. J. Chem. Phys. 2000, 113, 9901.

(52) Henkelman, G.; Jonsson, H. J. Chem. Phys. 2000, 113, 9978.

(53) Chu, J. W.; Trout, B. L.; Brooks, B. R. J. Chem. Phys. 2003, 119, 12708.

(54) Maragakis, P.; Andreev, S. A.; Brumer, Y.; Reichman, D. R.; Kaxiras, E.*J. Chem. Phys.* 2002, *117*, 4651.

- (55) Granot, R.; Baer, R. J. Chem. Phys. 2008, 128.
- (56) Ghasemi, S. A.; Goedecker, S. J. Chem. Phys. 2011, 135.
- (57) Frisch, M. J.; Trucks, G. W.; Schlegel, H. B.; Scuseria, G. E.; Robb, M.

A.; Cheeseman, J. R.; Scalmani, G.; Barone, V.; Mennucci, B.; Petersson, G. A.;

Nakatsuji, H.; Caricato, M.; Li, X.; Hratchian, H. P.; Iszmaylov, A. F.; Bloino, J.; Zheng,

G.; Sonnenberg, J. L.; Hada, M.; Ehara, M.; Toyota, K.; Fukuda, R.; Hasegawa, J.; Ishida,

M.; Nakajima, T.; Honda, Y.; Kitao, O.; Nakai, H.; Vreven, T.; Montgomery Jr., J. A.;

Peralta, J. E.; Ogliaro, F.; Bearpark, M.; Heyd, J. J.; Brothers, E.; Kudin, K. N.;

Staroverov, V. N.; Kobayashi, R.; Normand, J.; Raghavachari, K.; Rendell, A.; Burant, J.

C.; Iyengar, S. S.; Tomasi, J.; Cossi, M.; Rega, N.; Millam, J. M.; Klene, M.; Knox, J. E.;

Cross, J. B.; Bakken, V.; Adamo, C.; Jaramillo, J.; Gomperts, R.; Stratmann, R. E.;

Yazyev, O.; Austin, A. J.; Cammi, R.; Pomelli, C.; Ochterski, J. W.; Martin, R. L.;

Morokuma, K.; Zakrzewski, V. G.; Voth, G. A.; Salvador, P.; Dannenberg, J. J.;

Dapprich, S.; Daniels, A. D.; Farkas, O.; Foresman, J. B.; Ortiz, J. V.; Cioslowski, J.;

Fox, D. J.; Gaussian Inc.: Wallingford CT, 2009.

(58) Chapter 3.

- (59) Chapter 2.
- (60) Chapter 2.
- (61) Chapter 3.
- (62) Peng, C. Y.; Schlegel, H. B. Isr. J. Chem. 1993, 33, 449.
- (63) Chapter 2.
- (64) Goerigk, L.; Grimme, S. J. Chem. Theory Comp. 2010, 6, 107.
- (65) Zhao, Y.; Gonzalez-Garcia, N.; Truhlar, D. G. J. Phys. Chem. A 2005, 109,

2012.

- (66) Zhao, Y.; Truhlar, D. G. J. Phys. Chem. A 2004, 108, 6908.
- (67) Xu, X. F.; Alecu, I. M.; Truhlar, D. G. J. Chem. Theory Comp. 2011, 7, 1667.
  - (68) Guner, V.; Khuong, K. S.; Leach, A. G.; Lee, P. S.; Bartberger, M. D.;

Houk, K. N. J. Phys. Chem. A 2003, 107, 11445.

- (69) Riley, K. E.; Op't Holt, B. T.; Merz, K. M. J. Chem. Theory Comp. 2007, 3, 407.
  - (70) Herrera, B.; Toro-Labbe, A. J. Phys. Chem. A 2004, 108, 1830.
  - (71) Guevara-Garcia, A.; Echegaray, E.; Toro-Labbe, A.; Jenkins, S.; Kirk, S.

R.; Ayers, P. W. J. Chem. Phys. 2011, 134, 234106.

(72) Jaque, P.; Correa, J. V.; De Proft, F.; Toro-Labbe, A.; Geerlings, P.

Canadian Journal of Chemistry-Revue Canadienne De Chimie 2010, 88, 858.

(73) Mitra, S.; Chandra, A. K.; Gashnga, P. M.; Jenkins, S.; Kirk, S. R. J. Mol.*Model.* 2012, 18, 4225.

(74) Coyle, J. P.; Johnson, P. A.; DiLabio, G. A.; Barry, S. T.; Muller, J. *Inorg. Chem.* **2010**, *49*, 2844.

- (75) S. Rabi, S. K. Burger, and P. W. Ayers, in preparation.
- (76) Liu, Y.; Ayers, P. W. J. Math. Chem. 2011, 49, 1291.
- (77) Kabsch, W. Acta Crystallogr., Sect. A: Found. Crystallogr. 1976, 32, 922.

Chapter 5

# Systematic Assessment of Quantum-Chemical Initial Hessian Approximations for Transition-State Optimization

### 5.1 Overview

The effects of the initial Hessian calculation on the performance of three different quasi-Newton-based approaches for transition-state optimization is assessed using a recently presented database of 131 reactions. All three methods (using the exact Hessian, using the Hessian from a Hartree-Fock calculation with a minimal basis set, and using the Hessian from the AM1 semiempirical method) suffice. Since Hartree-Fock with a minimal basis set performs just as well as using the exact Hessian, the low cost of this initial approximation suggests that it should be used more broadly in geometry optimization methods.

### 5.2 Introduction

The initial approximation to the second-derivative matrix (the Hessian) can greatly affect the efficiency of quasi-Newton approaches for molecular structure optimization.<sup>1-5</sup> Partly because of this, there has been significant work on developing reasonably accurate, yet computationally inexpensive, approximations to the initial Hessian. In geometry minimization, the initial Hessian approximation is less critical, and methods based on molecular mechanics force fields are often employed.<sup>1,2,4,6-8</sup> For transition-state optimization, valence force fields are often inadequate,<sup>4</sup> and it is common to use an initial Hessian from quantum chemistry calculations, most frequently by initially computing the Hessian exactly, but using a quasi-Newton method thereafter.<sup>2,4</sup>

Because exact initial computations of the Hessian can be computationally expensive, we wanted to investigate whether other, less computationally demanding, quantum chemistry calculations would suffice. Specifically, we will assess how the performance of three transition state optimizers for three different choices of the initial Hessian. The transition-state optimizers we consider are the standard Berny optimizer<sup>3,9</sup> from the *Gaussian* program, the *Saddle* optimizer we recently developed,<sup>10</sup> and the QST3 method<sup>11</sup>. The initial Hessian is computed using either the semiempirical AM1 method,<sup>12-14</sup> Hartree-Fock with a minimal basis set (STO-3G), or the exact initial Hessian (HF/6-31++G(d,p). Calculations using the Berny and QST3 methods were completed using the *Gaussian* program;<sup>15</sup> the *Saddle* optimizations were performed with our in-house code, using energies and gradients from *Gaussian*.

All three optimizers use the Bofill94 quasi-Newton update<sup>16</sup> together with a trustradius approach for selecting steps. In the Berny optimizer, steps are made using the rational-function approximation, which strives to include the effects of higher-order derivatives.<sup>17-19</sup> Our *Saddle* optimizer is based on the realization that chemical transitions can be described with a reduced-dimensionality energy surface.<sup>20-22</sup> In *Saddle*,<sup>23</sup> elements of the Hessian that involve the key chemical coordinates (e.g., the interatomic distances between the atoms involved in bond-formation and bond-cleavage) are calculated accurately with finite differences,<sup>5</sup> while the other elements of the Hessian are estimated using a normal quasi-Newton method. QST3 also seeks to use information about the key chemical coordinates;<sup>11</sup> it uses the reactant and product structures to provide an initial refinement to the transition-state structure; after this initial stage, it reverts to a standard (in this case, Berny) transition-state optimizer.

### 5.3 Testing Procedure

To test the effect of the initial Hessian approximation, we considered 128 reactions from the database of reactions that we recently compiled. This reaction database includes the usual Baker-Chan set,<sup>24</sup> as well as reactions used for testing density-functional theory approximations<sup>25</sup> and reactions from other sources,<sup>5,26-31</sup> including about thirty reactions from our own unpublished research. All calculations were performed using the *Gaussian* program, with HF/6-31++G(d,p). Three reactions involving copper atoms were omitted here, because the AM1 parameters are not available for copper. The list of reactions, together with the reactant, product, and transition-state geometries we use, are provided as supplementary material.

We considered fourteen different initial guesses for each reaction. Eleven of these guesses were obtained by randomly perturbing the system away from the transition state using procedures we have previously presented.<sup>32</sup> The first of these perturbations is a random perturbation of the Cartesian coordinates of the atomic nuclei, so that the root-mean-square displacement of the nuclei is  $\varepsilon = 0.1$  a.u.. (The distance between the perturbed and unperturbed structures is measured by aligning the structures using the Kabsch algorithm,<sup>33</sup> then measuring the Cartesian distance between the structures.) As noted in Chapter 4, this perturbation sometimes generates chemically unreasonable transition-state guesses, but it is interesting to verify whether various approaches are capable of addressing these challenging initial guesses. To make a more chemically reasonable initial guess, we identified the key chemical coordinates involved in the transformation; we then randomly perturb these coordinates so that the distance of the

perturbed structure from the exact transition-state structure is  $\varepsilon = 0.4$  a.u.. In this work, we consider one "good guess" (where all atomic coordinates were displaced, with  $\varepsilon = 0.1$ a.u.) and ten "bad guesses" (where only the reduced coordinates were displaced, by  $\varepsilon = 0.4$  a.u.). The second set of ten guesses are representative of the typical "bad guesses" for a transition-state geometry that one might manually prepare using a graphical user interface (GUI).

We also consider three additional guesses using a transition-state-guessing method we recently developed.<sup>34</sup> These guesses are constructed by finding the molecular configuration that is as close as possible to the average of the internal coordinates from the reactant and product geometries, which we denote as  $\mathbf{q}^{(\text{reactant})}$  and  $\mathbf{q}^{(\text{product})}$ , respectively. Specifically, the three deterministic transition-state guesses are:

$$\mathbf{x}^{(\text{Method I})} = \arg \max_{p} \min_{\mathbf{x}} \left| \mathbf{q}(\mathbf{x}) - \left[ (1-p) \mathbf{q}^{(\text{reactant})} + p \mathbf{q}^{(\text{product})} \right] \right|^2$$
(5.1)

$$\mathbf{x}^{(\text{Method 2})} = \arg\min_{\mathbf{x}} \left( \frac{1}{2} \left| \mathbf{q}(\mathbf{x}) - \mathbf{q}^{(\text{reactant})} \right| + \frac{1}{2} \left| \mathbf{q}(\mathbf{x}) - \mathbf{q}^{(\text{product})} \right| \right)^2$$
(5.2)

$$\mathbf{x}^{(\text{Method 3})} = \arg\min_{\mathbf{x}} \left( \frac{1}{2} \left| \mathbf{q}(\mathbf{x}) - \mathbf{q}^{(\text{reactant})} \right|^2 + \frac{1}{2} \left| \mathbf{q}(\mathbf{x}) - \mathbf{q}^{(\text{product})} \right|^2 \right)$$
(5.3)

These guesses typically are slightly further from the true transition state than the "good guess," but significantly closer than the ten "bad guesses." However, the "good guess" includes substantial displacements of internal coordinates that are totally unrelated to the reaction, which can lead a method to locate an incorrect transition state. The guesses prepared from Eqs. (5.1)-(5.3), however, tend to be very accurate for all internal coordinates that have almost the same value in the reactant and product geometry.

Altogether there are 1,792 initial guesses—fourteen initial guess for each of 128 reactions.

From a given initial guess structure, the Hessian is computed exactly, at the AM1, or at the HF/STO-3G level. We then attempt to optimize the transition state using the Berny optimizer<sup>3,9</sup> and the QST3 optimizer<sup>11</sup> in *Gaussian '09*,<sup>15</sup> or using our own *Saddle* transition-state optimizer (which uses energy and gradient calculations from *Gaussian*).<sup>35</sup> Calculations are considered to fail if the program crashes for any reason, if convergence is not achieved in 200 iterations, or if the program converges to the wrong transition state. The method is considered to have converged to the wrong transition state if the transition state differs from the energy of the targeted transition state by more than  $10^{-3}$  a.u. and if the interatomic distances in the vicinity of the reaction site have a weighted percentage error of more than 2%. (That is, we are using the same criteria for evaluating successful convergence that was used in Chapter 4.) Our results are not sensitive to the way we classify whether the program finds the correct transition state. We prefer this criterion because it allows minor conformational changes far from the reacting atoms.

### 5.4 Discussion

Table 5.1 reports the number of iterations required to converge from each class of initial guesses for the transition state geometry and each method of computing the initial Hessian. Reactions that fail to converge are not included in this average. Table 5.2 reports the percentage of transition-state guesses that converge for each method. The quality of the transition-state guessing methods from Chapter 2 is clear: transition state guesses

from Eqs. (5.1)-(5.3) converge more rapidly (in fewer iterations) and robustly (more frequently) to the correct transition state than the randomly generated guesses. The quality of our *Saddle* transition-state optimizer is also clear: while *Saddle* is somewhat more expensive than the other optimizers, it converges a significantly larger fraction of the calculations. (*Saddle*'s computational performance is better than one would infer from Table 5.1. It is handicapped by the fact it converges many more reactions, and converging to the correct transition state from a poor initial guess requires many additional gradient calculations. For reactions where all three optimizers converge, *Saddle* typically requires only a few additional gradient evaluations.)

Tables 5.1 and 5.2 show that, when optimizing transition states, a rather inexpensive *ab initio* calculation—Hartree-Fock with a minimal basis set—suffices to provide a sufficiently good initial guess for the Hessian. Indeed, for the methods we considered, using HF/STO-3G to compute the initial Hessian approximation, instead of computing the initial Hessian exactly, did not significantly change the number of iterations required. Furthermore, the percentage of reactions that successfully converged to the correct transition state is comparable when HF/STO-3G is used to compute the initial Hessian. It is difficult to establish a clear preference between constructing the initial Hessian exactly, or with HF/STO-3G: the best choice seems to depend on the specific choice of initial guess, and the specific transition-state optimizer being used. Given that HF/STO-3G initial Hessian, resorting to the exact initial Hessian only if one encounters problems converging to the targeted transition state.

We did not systematically study whether HF/STO-3G Hessian is a good initial guess for transition-state optimizations with density-functional theory and other methods. Based on limited experiments, we believe that HF/STO-3G is a suitable guess in most cases, but determining the Hessian at the same level of electronic theory, with the STO-3G basis, might work better. Establishing this would require further tests. It is noteworthy that Hartree-Fock calculations often give reliable geometries,<sup>36</sup> so the HF/STO-3G initial Hessian guess is a reasonable protocol even for transition-state optimizations using post-Hartree-Fock methods.

If even a minimal basis set Hartree-Fock calculations is unaffordable, the initial Hessian can be taken from a semiempirical calculation (here, AM1). Using AM1 to construct the initial Hessian gives consistently inferior results. However, while the AM1 Hessian gives noticeably worse results, it is not qualitatively inferior.

Based on these results, we recommend using the *Saddle* transition-state optimizer, with either the HF/STO-3G or the exact initial Hessian. Among widely available algorithms, the QST3 method, with the HF/STO-3G initial Hessian, is a good general-purpose approach.

#### Ph.D. Thesis – Sandra Rabi; McMaster University – Chemistry and Chemical Biology

**Table 5.1.** Average number of steps for each initial guess as the choice of the initial Hessian was varied. The results for the best

 choice of initial Hessian for each set of initial guesses, for each optimization method, is given in bold.

Method		Saddle	2	Standard Berny Algorithm			QST3 Algorithm		
Initial Hessian	AM1	HF/ STO-3G	Exact Hessian	AM1	HF/ STO-3G	Exact Hessian	AM1	HF/ STO-3G	Exact Hessian
Good Guess	19.8	17.1	17.2	29.7	17.6	22.9	32.4	20.0	27.3
Bad Guess	29.4	26.7	26.1	19.3	15.4	15.3	22.8	18.3	18.2
Method 1	13.3	12.5	11.8	20.4	12.6	10.7	22.8	17.0	9.1
Method 2	13.4	11.9	10.9	13.5	7.7	6.1	16.0	10.4	10.7
Method 3	16.8	16.4	17.4	18.7	10.0	9.0	24.7	16.5	13.5

**Table 5.2.** Percentage convergence of reactions for each initial guess as the choice of the initial Hessian was varied. The results for the

most robust choice of initial Hessian for each set of initial g	guesses, for each	optimization metho	od, is given in bold.
---	-------------------	--------------------	-----------------------

Method		Saddle		Standard Berny Algorithm		QST3 Algorithm			
Initial Hessian	AM1	HF/STO-3G	Exact Hessian	AM1	HF/STO-3G	Exact Hessian	AM1	HF/STO-3G	Exact Hessian
Good Guess	95.3%	96.1%	100.0%	84.4%	92.2%	87.5%	88.3%	92.2%	93.8%
Bad Guess	88.4%	89.8%	89.5%	52.0%	59.1%	59.6%	78.2%	78.8%	80.0%
Method 1	96.9%	96.9%	97.7%	78.9%	85.2%	84.4%	86.7%	92.2%	89.8%
Method 2	97.7%	97.7%	97.7%	87.5%	93.8%	93.8%	87.5%	93.0%	90.6%
Method 3	84.4%	85.9%	86.7%	75.8%	78.9%	81.3%	82.8%	89.8%	85.9%

### 5.5 References

- (1) Fischer, T. H.; Almlof, J. J. Phys. Chem. **1992**, *96*, 9768.
- (2) Schlegel, H. B. *Theor. Chim. Act.* **1984**, *66*, 333.
- (3) Peng, C. Y.; Ayala, P. Y.; Schlegel, H. B.; Frisch, M. J. J. Comput. Chem.

**1996**, *17*, 49.

- (4) Bakken, V.; Helgaker, T. J. Chem. Phys. 2002, 117, 9160.
- (5) Burger, S. K.; Ayers, P. W. J. Chem. Phys. 2010, 132, 234110.
- (6) Lindh, R.; Bernhardsson, A.; Karlstrom, G.; Malmqvist, P. A. Chem. Phys.

Lett. 1995, 241, 423.

- (7) Swart, M.; Bickelhaupt, F. M. Int. J. Quantum Chem. 2006, 106, 2536.
- (8) Rondinelli, J. M.; Deng, B.; Marks, L. D. Computational Materials

Science 2007, 40, 345.

- (9) Schlegel, H. B. J. Comput. Chem. **1982**, *3*, 214.
- (10) Chapter 3.
- (11) Peng, C. Y.; Schlegel, H. B. Isr. J. Chem. 1993, 33, 449.
- (12) Frisch, M.; Scalmani, G.; Vreven, T.; Zheng, G. S. Mol. Phys. 2009, 107,

881.

(13) Thiel, W.; Voityuk, A. A. J. Phys. Chem. 1996, 100, 616.

(14) Dewar, M. J. S.; Zoebisch, E. G.; Healy, E. F.; Stewart, J. J. P. J. Am.

Chem. Soc. 1985, 107, 3902.

(15) Frisch, M. J.; Trucks, G. W.; Schlegel, H. B.; Scuseria, G. E.; Robb, M.

A.; Cheeseman, J. R.; Scalmani, G.; Barone, V.; Mennucci, B.; Petersson, G. A.;

Nakatsuji, H.; Caricato, M.; Li, X.; Hratchian, H. P.; Iszmaylov, A. F.; Bloino, J.; Zheng,

G.; Sonnenberg, J. L.; Hada, M.; Ehara, M.; Toyota, K.; Fukuda, R.; Hasegawa, J.; Ishida,

M.; Nakajima, T.; Honda, Y.; Kitao, O.; Nakai, H.; Vreven, T.; Montgomery Jr., J. A.;

Peralta, J. E.; Ogliaro, F.; Bearpark, M.; Heyd, J. J.; Brothers, E.; Kudin, K. N.;

Staroverov, V. N.; Kobayashi, R.; Normand, J.; Raghavachari, K.; Rendell, A.; Burant, J.

C.; Iyengar, S. S.; Tomasi, J.; Cossi, M.; Rega, N.; Millam, J. M.; Klene, M.; Knox, J. E.;

Cross, J. B.; Bakken, V.; Adamo, C.; Jaramillo, J.; Gomperts, R.; Stratmann, R. E.;

Yazyev, O.; Austin, A. J.; Cammi, R.; Pomelli, C.; Ochterski, J. W.; Martin, R. L.;

Morokuma, K.; Zakrzewski, V. G.; Voth, G. A.; Salvador, P.; Dannenberg, J. J.;

Dapprich, S.; Daniels, A. D.; Farkas, O.; Foresman, J. B.; Ortiz, J. V.; Cioslowski, J.;

Fox, D. J.; Gaussian Inc.: Wallingford CT, 2009.

- (16) Bofill, J. M. J. Comput. Chem. 1994, 15, 1.
- (17) Banerjee, A.; Adams, N.; Simons, J.; Shepard, R. J. Phys. Chem. 1985, 89,

52.

- (18) Baker, J. J. Comput. Chem. 1986, 7, 385.
- (19) Simons, J.; Jorgensen, P.; Taylor, H.; Ozment, J. J. Phys. Chem. 1983, 87,

2745.

- (20) Bofill, J. M.; Anglada, J. M. Theor. Chem. Acc. 2001, 105, 463.
- (21) Burger, S. K.; Ayers, P. W. J. Chem. Theory Comp. 2010, 6, 1490.
- (22) Liu, Y. L.; Burger, S. K.; Dey, B. K.; Sarkar, U.; Janicki, M.; Ayers, P. W.

In Quantum Biochemistry; Matta, C. F., Ed.; Wiley-VCH: Boston, 2010.

(23) Chapter 3.

- (24) Baker, J.; Chan, F. R. J. Comput. Chem. 1996, 17, 888.
- (25) Goerigk, L.; Grimme, S. J. Chem. Theory Comp. 2010, 6, 107.
- (26) Herrera, B.; Toro-Labbe, A. J. Phys. Chem. A 2004, 108, 1830.
- (27) Guevara-Garcia, A.; Echegaray, E.; Toro-Labbe, A.; Jenkins, S.; Kirk, S.

R.; Ayers, P. W. J. Chem. Phys. 2011, 134, 234106.

(28) Jaque, P.; Correa, J. V.; De Proft, F.; Toro-Labbe, A.; Geerlings, P.

Canadian Journal of Chemistry-Revue Canadienne De Chimie 2010, 88, 858.

(29) Liu, Y.; Ayers, P. W. J. Math. Chem. 2011, 49, 1291.

(30) Coyle, J. P.; Johnson, P. A.; DiLabio, G. A.; Barry, S. T.; Muller, J. Inorg.

*Chem.* **2010**, *49*, 2844.

(31) Mitra, S.; Chandra, A. K.; Gashnga, P. M.; Jenkins, S.; Kirk, S. R. J. Mol.

Model. 2012, 18, 4225.

- (32) See Chapter 4.
- (33) Kabsch, W. Acta Crystallogr., Sect. A: Found. Crystallogr. 1976, 32, 922.
- (34) See Chapter 2.
- (35) Chapter 3.
- (36) Levy, M.; Perdew, J. P. J. Chem. Phys. 1986, 84, 4519.

### Chapter 6

# **Automatic Detection of Key Chemical Coordinates**

# for Reduced-Coordinate Approaches to

# **Transition-State Optimization**

### 6.1 Overview

Selecting an appropriate coordinate system and specifying key "reduced" coordinates that should be treated with extra care, is critical for robust transition-state optimization methods. We develop, and assess, several methods for selecting an appropriate system of redundant internal coordinates, and the key reduced coordinates, from the molecular structures of the reactant and product and guess for the transition-state structure. While it is better to select the key coordinates manually, using chemical intuition, computational approaches relying on automatic specification of coordinates still perform well.

### 6.2 Introduction

Because of the high dimensionality of molecular potential energy curves, it is often advisable to select several key chemical coordinates, which are then subjected to greater scrutiny. These key chemical coordinates, which typically correspond to internuclear distances of the bonds that fracture and form in the course of the chemical reaction, are called reduced chemical coordinates, and the potential (or free) energy surface written only as a function of these coordinates is a reduced potential energy surface. Reduced coordinates are used in dynamical and deterministic methods for exploring chemical reaction pathways,<sup>1-7</sup> and also to improve the efficiency of geometry optimization methods.<sup>8-10,11</sup> It is this latter problem, and specifically the use of reduced

coordinates to improve the robustness of transition-state optimization methods, that we focus on here.<sup>12</sup>

We have recently developed a transition-state optimization method in redundant internal coordinates,<sup>13-20,21</sup> where the elements of the Hessian associated with the reduced coordinates are accurately computed with finite differences.<sup>11</sup> This method, which we call *Saddle*, is slightly slower than the best conventional methods, mainly because it uses additional gradient evaluations to ensure that the elements of the Hessian involving the reduced coordinates are accurate. However, it is significantly more robust than conventional methods, with a much lower failure rate, and a significantly lower probability of converging to the wrong transition state.<sup>12</sup> Unfortunately, *Saddle* requires that the user provide a list of the reduced coordinates, using her chemical intuition. This is rarely difficult (we did not exert any special care in selecting the reduced coordinates), but it means that *Saddle* is not a true black-box transition-state optimizer.

To remedy this, we developed a protocol for the automatic selection of reduced coordinates, using the structure of the reactants, the products, and the guess for the transition-state structure. The idea is that internal coordinates that change a significant amount between these three structures should be treated as reduced coordinates.

# 6.3 Protocol for Automatically Selecting Reduced Coordinates

To select the reduced internal coordinates, we compare the internal coordinates for three pairs of structures: (1) reactant and product, (2) reactant and transition state, (3) product and transition state. If, for any of these pairs of structures, any internuclear distance changes by more than half the sum of the covalent radii of its constituent atoms, that internuclear distance is designated as a reduced bond. After all the reduced bonds have been identified, we examine angles where all three atoms contribute to a reduced bond. (This is almost always the angle between two reduced bonds, but it does not have to be.) If these angles differ by more than 30 degrees, then the angle is selected to be reduced. With our database of reactions, there was not enough data to identify a robust criterion for selecting which dihedral angles should be reduced coordinates; reduced dihedral angles must be specified by the user.

### 6.4 Testing Protocol

To test the efficiency of this method for automatically selecting the reduced coordinates, a thorough comparison study was performed using reactions from our test set of reactions where the reactant and product do not dissociate.<sup>12</sup> (That is, when one computes the intrinsic reaction coordinate, the final reactant and product structures do not consist of two dissociating fragments.) We were left with 60 reactions: reactions 6, 10-18, 20, 22-23, 28-33, 35, 37, 39-44, 48-49, 52, 54-61, 63-67, 74, 82-84, 91, 94-95, 98-99,

102-105, 120, and 127-129 from reference 12. All the calculations were performed at the HF/6-31++G(p,d) level, using the *Gaussian* program.<sup>22</sup>

To test the method, we generated a variety of initial guesses, ranging from very good initial guesses to extremely poor initial guesses. Specifically, we interpolated between the reactant and product structures, in redundant internal coordinates, using two of the methods from reference 21,

$$\mathbf{x}^{(\text{Method 1})}(p) = \arg \min_{\mathbf{x}} \left| \mathbf{q}(\mathbf{x}) - \left[ (1-p) \mathbf{q}^{(\text{reactant})} + p \mathbf{q}^{(\text{product})} \right] \right|^{2}$$
(6.1)  
$$\mathbf{x}^{(\text{Method 3})}(p) = \arg \min_{\mathbf{x}} \left( (1-p) \left| \mathbf{q}(\mathbf{x}) - \mathbf{q}^{(\text{reactant})} \right|^{2} + p \left| \mathbf{q}(\mathbf{x}) - \mathbf{q}^{(\text{product})} \right|^{2} \right)$$
(6.2)

Here  $\mathbf{x}$  denotes the Cartesian coordinate of the atoms and  $\mathbf{q}$  denotes the molecular geometry in redundant internal coordinates.

The approximate reaction pathways,  $\mathbf{x}(p)$ , were divided into eight equivalent segments using the following values of p,

$$p^{(Method 1)}(n) = 0.125n$$
  $n = 0, 1, 2, \dots 8$  (6.3)

$$p^{(\text{Method }3)}(n) = \frac{1 - \sqrt{\left(\frac{1}{0.125n} - 1\right)}}{2 - \frac{1}{0.125n}} \qquad n = 0, 1, 2, \dots 8$$
(6.4)

When n = 4, the indeterminate form in Eq. (6.4) has the value  $p^{(Method 3)}(4) = 0.5$ . We used the seven structures between the reactant and the product, n = 1, 2, ..., 7, as initial guesses for the transition-state structure. Clearly the structures from midpoint of the

interpolant, n = 4, will be better guesses for the transition-state structure, on average, than those near the reactant (n < 4) or product (n > 4) structures.

The structures considered here are considerably worse than most of the structures we considered in reference 21 (where the reactant and product were set to the endpoints of a standard IRC calculation, and not the bottoms of the associated wells of the molecular potential energy surface) and the randomly generated structures in reference 12. For this reason, the tests considered here represent an extremely challenging test for transition-state optimizers. In particular, many of the initial guesses for the transition-state structure resemble the reactant or product structures much more strongly than they resemble the structure of the true transition state.

We considered four different ways to construct the redundant internal coordinates and to select the reduced coordinates.

- **user+TS.** The user specifies the reduced coordinates, and the system of redundant coordinates is determined from the transition-state guess. The resulting calculations use the version of *Saddle* presented in refs. [11-12].
- **user+all.** The user specifies the reduced coordinates and the initial structures for the transition state, reactant, and product. Redundant coordinates are constructed for the reactant, product, and transition-state structures and the union of these coordinates is used as the redundant internal coordinate system for the calculation.

- **auto+all.** The reduced coordinates are automatically generated using the protocol described in the first paragraph of section 6.3. Redundant coordinates are constructed for the reactant, product, and transition-state structures and the union of these coordinates is used as the redundant internal coordinate system for the calculation.
- both+all. The reduced coordinates are automatically generated, then the list of reduced coordinates is amended by any user-specified coordinates that were not among the automatically generated coordinates. The redundant coordinate system is the union of the redundant coordinate systems of the reactant, product, and transition state structures.

We used the same version of *Saddle* that was presented in 11, and systematically tested in 12, to try to optimize the transition state for these choices of coordinates, with all the initial transition state structures (n = 1, 2, ... 7). The initial Hessian was computed exactly. The average number of gradient evaluations required to locate the correct transition state (Table 6.1), along with the percentage of reactions that converge to the correct transition state (Table 6.2), were computed. As in refs. 12 and 23, a transition-state optimization is considered to fail if the program crashes for any reason (including SCF convergence failure), if convergence is not achieved in 200 iterations, or if the program converges to the wrong transition state. A transition state is considered to be incorrect if it differs in energy from the targeted transition state by more than  $10^{-3}$  a.u. and if the interatomic distances in the vicinity of the reaction site have a weighted percentage error of greater than 2%.

**Table 6.1.** Average number of steps for the various methods for choosing the reducedcoordinates (**user**-specified, **auto**matically generated, or **both** (the union of the two sets) and the redundant internal coordinates (using only the **TS** (transition-state structure), or using **all** available structures (the transition-state structure, the reactant structure, and the product structure)).

	<b>Type of Reduced Coordinates – Choice of Redundant Internals</b>							
n	user+TS user+all		auto+all	both+all				
1	56.3	45.4	42.8	44.7				
2	33.5	38.3	38.9	39.6				
3	33.1	32.5	31.8	31.5				
4	29.9	29.9	31.9	29.3				
5	36.2	33.0	33.0	28.8				
6	28.1	29.3	31.3	35.9				
7	46.6	39.5	45.2	41.6				
Average								
$(n = 1, 2, \dots, 7)$	36.3	34.5	35.0	34.3				

**Table 6.2.** Percentage convergence for the various methods for choosing the reducedcoordinates (**user**-specified, **auto**matically generated, or **both** (the union of the two sets) and the redundant internal coordinates (using only the **TS** (transition-state structure), or using **all** available structures (the transition-state structure, the reactant structure, and the product structure)).

	<b>Type of Reduced Coordinates – Choice of Redundant Internals</b>							
n	user+TS	user+all	auto+all	both+all				
1	47.5%	42.5%	31.7%	30.0%				
2	64.2%	63.3%	55.0%	46.7%				
3	78.3%	77.5%	70.0%	66.7%				
4	80.8%	77.5%	74.2%	71.7%				
5	87.5%	81.7%	78.3%	73.3%				
6	65.0%	63.3%	61.7%	65.0%				
7	53.3%	48.3%	36.7%	41.7%				
Average								
( <i>n</i> =								
1,2,7)	68.1%	64.9%	58.2%	56.4%				

**Figure 6.1.** The percentage of reactions that converged in a specified number of steps, using an initial structure taken from the midpoint of the approximate reaction path from Eqs. (6.1) and (6.2), depending on the choice of reduced and redundant internal coordinates.



### 6.5 **Results and Discussion**

The first column (**user+TS**) of Table 6.1 and Table 6.2 is the traditional transition-state optimization protocol: the reduced coordinates are specified by the user, and the redundant internal coordinates are constructed from the initial transition-state structure. The method converges relatively robustly (around 80% convergence), and relatively quickly (30 to 40 gradient evaluations) for guesses near the midpoint of the

approximate reaction paths from Eqs. (6.1) and (6.2). For guesses near the reactant or product structures, the number of iterations rises, and the percentage of calculations that converge falls to about 50%.

If the reactant and product structures are available, we can use these structures to construct redundant internal coordinate systems; usually these systems will be slightly different from the redundant internal coordinate system from the transition-state structure. If we take the union of these coordinates (user+all), one might expect that one would have an even better system of redundant internal coordinates. As seen in tables 6.1 and 6.2, and in the performance plot for p = 0.5 (Figure 6.1), this does not seem to be the case. In most cases, calculations converge slightly quicker using the enriched coordinate system, but this is mainly because some of the most challenging cases fail to converge at all. Still, the method is relatively robust (around 80% convergence) for guesses near the transition-state. Using all three structures to construct the set of redundant internal coordinates is important, however, for initial structures near the reactant or the product (n= 1 and n = 7, respectively). In these cases, using the redundant internal coordinates that describe the reactant and product structures is helpful for the early stages of the optimization, and it greatly increases the speed of the optimization, while decreasing the percentage of reactions that converge by less than 5%. Overall it seems that having a coordinate system that is "too redundant" decreases the robustness of the method, but increases its speed.

Next we tested the automatically generated reduced coordinate protocol from section 6.3 (**auto+all**). This tends to generate more reduced coordinates than would be

223

generated by intuition. Because the number of reduced coordinates is larger, a greater fraction of the Hessian is accurately approximated by finite differences in this case. The speed of the method is comparable, but significantly fewer reactions converge, especially when the guessed structure is far from the true transition-state. *Saddle* forces the reduced-coordinate block of the Hessian to have exactly one imaginary frequency; this helps guide the answer to the transition state. This is more effective, however, when the reduced block of the Hessian is small; when the reduced block of the Hessian is larger, it is more likely that the procedure assigns an imaginary frequency to the wrong normal mode, which can lead the method to converge to the wrong transition state, or cause convergence failure.

We noticed that occasionally the automatic selection of reduced coordinates failed to select a coordinate that we had manually identified as important. This led us to try adding the user-specified reduced coordinates to the automatically selected reduced coordinates (**both+all**). The resulting calculations converge acceptably quickly, but converge less frequently than the analogous cases, where either only the user-specified reduced coordinates (**user+all**) or automatically-selected reduced coordinates (**auto+all**) were used. This is probably because making an even larger reduced space makes it even less likely that correct eigenvector of the reduced block of the Hessian will be selected to correspond to a negative frequency.

For guesses near the true transition state (n = 3,4,5), all of the methods we considered work well, converging, on average, in 30-40 iterations. Calculations using user-specified reduced coordinates converge about 80% of the time. Calculations using

224

automatically-specified reduced coordinates converge about 70% of the time. For initial guesses that are further from the transition state, it is more important for the user to specify the reduced coordinates by hand.

While these conclusions are specific to the *Saddle* program, our procedure for selecting reduced coordinates automatically can obviously be applied to other reduced-coordinate methods.<sup>1-10</sup> In addition, our results suggest that if one is struggling to converge to a transition-state using an optimization method based on redundant internal coordinates, it is sometimes beneficial to use the union of the redundant internal coordinate systems of the transition-state, reactant, and product structures.
# 6.6 References:

(1) Laio, A.; Parrinello, M. Proc. Natl. Acad. Sci. 2002, 99, 12562.

(2) Ensing, B.; Laio, A.; Gervasio, F. L.; Parrinello, M.; Klein, M. L. J. Am.

Chem. Soc. 2004, 126, 9492.

- (3) Wu, Y. D.; Schmitt, J. D.; Car, R. J. Chem. Phys. 2004, 121, 1193.
- (4) Dey, B. K.; Janicki, M. R.; Ayers, P. W. J. Chem. Phys. 2004, 121, 6667.

(5) Burger, S. K.; Liu, Y. L.; Sarkar, U.; Ayers, P. W. J. Chem. Phys. 2009,

130, 024103.

- (6) Burger, S. K.; Ayers, P. W. J. Chem. Theory Comp. 2010, 6, 1490.
- (7) Dey, B. K.; Ayers, P. W. *Mol. Phys.* **2006**, *104*, 541.
- (8) Bofill, J. M.; Anglada, J. M. *Theor. Chem. Acc.* **2001**, *105*, 463.
- (9) Anglada, J. M.; Besalu, E.; Bofill, J. M.; Crehuet, R. J. Comput. Chem.

2001, 22, 387.

- (10) Burger, S. K.; Ayers, P. W. J. Chem. Phys. 2010, 132, 234110.
- (11) Chapter 3
- (12) Chapter 4
- (13) Fogarasi, G.; Zhou, X. F.; Taylor, P. W.; Pulay, P. J. Am. Chem. Soc.

**1992**, *114*, 8191.

- (14) Pulay, P.; Fogarasi, G. J. Chem. Phys. 1992, 96, 2856.
- (15) Baker, J.; Kessi, A.; Delley, B. J. Chem. Phys. 1996, 105, 192.
- (16) Baker, J.; Pulay, P. J. Chem. Phys. 1996, 105, 11100.
- (17) Baker, J.; Kinghorn, D.; Pulay, P. J. Chem. Phys. 1999, 110, 4986.

(18) von Arnim, M.; Ahlrichs, R. J. Chem. Phys. 1999, 111, 9183.

(19) Peng, C. Y.; Ayala, P. Y.; Schlegel, H. B.; Frisch, M. J. J. Comput. Chem.**1996**, 17, 49.

- (20) Bakken, V.; Helgaker, T. J. Chem. Phys. 2002, 117, 9160.
- (21) Chapter 2
- (22) Frisch, M. J.; Trucks, G. W.; Schlegel, H. B.; Scuseria, G. E.; Robb, M.

A.; Cheeseman, J. R.; Scalmani, G.; Barone, V.; Mennucci, B.; Petersson, G. A.;

Nakatsuji, H.; Caricato, M.; Li, X.; Hratchian, H. P.; Iszmaylov, A. F.; Bloino, J.; Zheng,

G.; Sonnenberg, J. L.; Hada, M.; Ehara, M.; Toyota, K.; Fukuda, R.; Hasegawa, J.; Ishida,

M.; Nakajima, T.; Honda, Y.; Kitao, O.; Nakai, H.; Vreven, T.; Montgomery Jr., J. A.;

Peralta, J. E.; Ogliaro, F.; Bearpark, M.; Heyd, J. J.; Brothers, E.; Kudin, K. N.;

Staroverov, V. N.; Kobayashi, R.; Normand, J.; Raghavachari, K.; Rendell, A.; Burant, J.

C.; Iyengar, S. S.; Tomasi, J.; Cossi, M.; Rega, N.; Millam, J. M.; Klene, M.; Knox, J. E.;

Cross, J. B.; Bakken, V.; Adamo, C.; Jaramillo, J.; Gomperts, R.; Stratmann, R. E.;

Yazyev, O.; Austin, A. J.; Cammi, R.; Pomelli, C.; Ochterski, J. W.; Martin, R. L.;

Morokuma, K.; Zakrzewski, V. G.; Voth, G. A.; Salvador, P.; Dannenberg, J. J.;

Dapprich, S.; Daniels, A. D.; Farkas, O.; Foresman, J. B.; Ortiz, J. V.; Cioslowski, J.;

Fox, D. J.; Gaussian Inc.: Wallingford CT, 2009.

(23) Chapter 5

Chapter 7

**Conclusions & Future Work** 

# 7.1 Summary

This thesis presents new methods for transition-state optimization. While efficient transition-state optimizers already exist, they are somewhat unreliable. Our aim was to invent a new transition-state optimization method that was much more robust than existing methods, but not much (if any) slower. The method presented in chapter 3, and tested systematically in chapter 4, achieves this objective.

In order to determine a chemical reaction mechanism, one must find the lowest (or several of the lowest) energy pathway from the reactant to the product on the molecular potential energy surface. The highest point on this path is called the transition state; it represents the bottleneck between the reactant and product structures. From knowledge of the transition-state, one may estimate the rate of reaction along a given reaction pathway using traditional transition-state theory. Transition-state theory, together with straightforward corrections for tunnelling, barrier-recrossing, etc., is sufficiently accurate for most computational reaction modeling, partly because the approximations inherent in transition-state theory are usually smaller than other approximations—neglect or approximation of thermal effects and the molecular environment (e.g., solvation), approximate values for electronic energies, etc.—that are commonly made.

Traditionally, quantum chemists have been primarily concerned with computing the value of the molecular potential energy surface for a given molecular structure. However, now that reliable, accurate, and computationally efficient software for evaluating molecular energies is available, the biggest challenge in computational modeling of chemical reactions is finding transition states. Transition-state optimization is an old problem, with modern transition-state optimization methods appearing in the late 1970's. Our work may be viewed as a refinement of these venerable algorithms.

Transition-state optimization methods are extremely sensitive to the initial guess for the transition state structure. Almost any reasonable algorithm will work well if the initial guess is good enough, but even the best algorithms will fail if the initial guess is very poor. In Chapter 2, we designed a new method for predicting the transition-state structure, using synchronous transit paths in redundant internal coordinates. The idea is that the transition state structure is approximately a weighted average of the reactant and product structures, in redundant internal coordinates. Tested against a large database of reactions that we compiled for this purpose, the approaches we designed have surprising accuracy of about .1 a.u. root-mean-square deviation from the correct transition state.

In order to make the transition-state guessing algorithm in chapter 2 robust, we developed new tools for working with redundant internal coordinates. The most important revision was the introduction of robust dihedral descriptors; these new descriptors replace the conventional dihedral angle for measuring molecular torsions. However, unlike the conventional dihedral angle, these descriptors do not lead to numerical ill-conditioning when the atoms involved in the torsion are nearly linear. Second, we proposed the manifold-projection method for converting between redundant internal coordinates and Cartesian coordinates. The idea of manifold projection is simple: given a set of redundant internal coordinates, which may or may not be physically realizable by any molecular geometry, select the molecular geometry that comes as close as possible to reproducing the desired coordinates. The Cartesian coordinates corresponding to this molecular

geometry are then used to evaluate the molecular potential energy and its gradient by an electronic structure theory method.

These tools for working with redundant internal coordinates made it possible to develop the more robust transition-state optimization method, called *Saddle*, presented in Chapter 3. *Saddle* is based on the idea that the molecular rearrangements in chemical reactions can be described, qualitatively, using only a few internal coordinates (e.g., the lengths of the breaking and forming bonds, plus the angles between them). These key coordinates define a reduced-dimensionality potential energy surface; transition-state optimizers that treat the reduced coordinates with extra care are expected to be more robust than traditional approaches. Our idea is to use finite-differences, as necessary, to ensure that the components of the Hessian involving the reduced coordinates are accurate. This gives an algorithm that is more robust, but not much more costly, than traditional approaches.

In the process of developing *Saddle*, we designed new quasi-Newton updates that are appropriate for systems containing reduced coordinates. We also designed a gradientbased trust region method that seems more appropriate for optimizations to stationary points, instead of extrema.

The impressive performance of *Saddle*, relative to the traditional approaches implemented in the *Gaussian* program, is confirmed in chapter 4. Chapter 4 also presents a new protocol for testing transition-state optimization methods. The key components of this algorithm are a test-set of chemical reactions (an extension of the set in chapter 2), together with a procedure for creating random guesses for their transition states that are a

specified distance from the correct structure. By evaluating how different methods perform as one gets further and further from the correct structure (as the initial guesses become worse and worse), we can assess their relative performance in terms of computational cost and robustness. *Saddle* is slightly more costly than existing methods, but it is significantly more robust; it works much better for very poor initial guesses than traditional methods.

Chapters 5 and 6 are refinements of the *Saddle* algorithm. Chapter 5 shows that one does not need to compute the exact initial Hessian; an inexpensive Hessian computed with the Hartree-Fock method, using a minimal basis set, works just as well. This insight extends to other transition-state finders. Chapter 6 investigates whether *Saddle* can be made into a black-box method, where the user does not need to manually identify the key reduced coordinates. The protocol described there requires the structures of the reactant, product, and transition-state guess; it identifies larger sets of reduced coordinates than a typical human user would choose, but the method converges nearly as well as manually selected reduced coordinates.

# 7.2 Future Work and Other Projects Not Included in the Thesis

#### A. Overview

In addition to the research presented here, I have been involved in many other projects during my Ph.D. years. These projects range from ideas for future work that I have formulated but not yet pursued, to projects for which preliminary results are available, to completed projects that I chose not to include in the thesis. (There are two published papers,<sup>1,2</sup> one accepted paper,<sup>3</sup> and a paper to be submitted to J. Mol. Modeling that are not included in the thesis.<sup>4</sup>) In the remainder of this document, I will discuss these projects, starting with ones related to the thesis itself (section 7.2.B), concluding with peripheral projects that I investigated at various points during my graduate studies (section 7.2.C).

## **B.** Extensions and Further Development of the Topics Covered in the Thesis

## 1. Improved Initial Guesses for Transition-State Geometries

In Chapter 2, three different ways of interpolating between the reactant and product structure in redundant internal coordinates were considered. For example,

$$\mathbf{x}(p) = \arg \min_{\mathbf{x}} \left| \mathbf{q}(\mathbf{x}) - \left[ (1-p) \mathbf{q}^{(\text{reactant})} + p \mathbf{q}^{(\text{product})} \right] \right|^2$$
(7.1)

Typically one selects the midpoint on the path (p = 0.5), or the value of p for which the value at the minimum is largest, as the initial guess for the transition state. However, we observed that using a better guess for the transition state gives faster convergence. For example, the point on the interpolated pathway, (7.1), that is closest to the transition state is

$$p_{\text{closest}} = \arg \min_{p} \left| \mathbf{x}(p) - \mathbf{x}_{\text{t.s.}} \right|^2$$
(7.2)

Determining  $p_{\text{closest}}$  is impossible without already knowing the transition state. However, if one knows certain information about the reactant and product, a *model* for the transition state can be constructed. For example, from the energy of the reactant and product, one can use the Hammond postulate to estimate whether the transition state will be closer to

the reactant (exothermic reactions,  $p_{closest} < 0.5$ ) or the product (endothermic reactions,  $p_{closest} > 0.5$ ). Together with Santa Rabi and Christopher Haddad, I built a model for  $p_{closest}$ . It will be interesting to test this model to see whether it improves the quality of the initial guess for transition-state optimization.

### 2. Alternative Secant Conditions for *Saddle*

In the appendix of chapter 3, three versions of the secant condition are presented. (Eqs. (85), (88), and (90)). Eq. (90) is a symmetrized version of Eq. (88); there is also a symmetrized version of Eq. (85).) We should implement these secant conditions and test to see which is the most accurate.

### 3. Strict Separation of Reduced and Nonredundant Coordinates

In order to make the secant condition reliable, after each step, *Saddle* aligns the nonredundant delocalized orthogonal internal coordinates (the **v**-space) so that they are maximally aligned to the previous step. In *Saddle* we assumed that the most important thing was to maximize the alignment between the basis vectors of the old and new **v**-spaces, as this should improve the reliability of the secant approximation. However, this has the unpleasant feature of mixing the reduced coordinates with the non-reduced coordinates at each step.

It might be better to prevent this mixing. To do this, one needs to align the reduced and non-reduced elements separately. To do this, one defines a reduced and nonreduced portion of the **V** matrix,

$$\mathbf{V}_{\text{reduced}} = \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 & \cdots & \mathbf{v}_R \end{bmatrix}$$

$$\mathbf{V}_{\text{nonreduced}} = \begin{bmatrix} \mathbf{v}_{R+1} & \mathbf{v}_{R+2} & \cdots & \mathbf{v}_{3N_{\text{atoms}} - 6(5)} \end{bmatrix}$$
(7.3)

and aligns the spaces separately. This requires performing two separate singular value decompositions,

$$\left(\tilde{\mathbf{V}}_{\text{reduced}}^{(\text{new})}\right)^{T} \mathbf{V}_{\text{reduced}}^{(\text{old})} = \mathbf{U}_{\text{reduced}} \boldsymbol{\Sigma}_{\text{reduced}} \mathbf{W}_{\text{reduced}}^{T}$$

$$\left(\tilde{\mathbf{V}}_{\text{nonreduced}}^{(\text{new})}\right)^{T} \mathbf{V}_{\text{nonreduced}}^{(\text{old})} = \mathbf{U}_{\text{nonreduced}} \boldsymbol{\Sigma}_{\text{nonreduced}} \mathbf{W}_{\text{nonreduced}}^{T}$$
(7.4)

The maximally aligned vectors, subject to the constraint that the nonreduced and reduced spaces do not mix, are

$$\mathbf{V}_{\text{reduced}}^{(\text{new})} = \tilde{\mathbf{V}}_{\text{reduced}}^{(\text{new})} \mathbf{U}_{\text{reduced}} \mathbf{W}_{\text{reduced}}^{T}$$

$$\mathbf{V}_{\text{nonreduced}}^{(\text{new})} = \tilde{\mathbf{V}}_{\text{nonreduced}}^{(\text{new})} \mathbf{U}_{\text{nonreduced}} \mathbf{W}_{\text{nonreduced}}^{T}$$
(7.5)

#### 4. Nonredundant Alternatives to Delocalized Internal Coordinates

The *Saddle* program uses a basis of nonredundant and orthogonal coordinates that is closely related to the delocalized internal coordinates proposed by Baker.<sup>5</sup> The difference is that we have to construct reduced and nonreduced subspaces.

Denote the elements of the Wilson **B** matrix as

$$b_{ij} = \frac{\partial q_i}{\partial x_j} \tag{7.6}$$

where  $q_i$  is a redundant internal coordinate and  $x_j$  is a Cartesian coordinate. We project the reduced coordinates,  $\hat{\mathbf{s}}^{(1)}, \hat{\mathbf{s}}^{(2)}, \dots, \hat{\mathbf{s}}^{(R)}$ , to construct "true" reduced coordinates, which are realizable with physical displacements of the atoms,

$$\mathbf{r}^{(k)} = \mathbf{B}\mathbf{B}^{+}\hat{\mathbf{s}}^{(k)}$$
  $k = 1, 2, \dots, R$  (7.7)

where  $\mathbf{B}^+$  denotes the Moore-Penrose pseudoinverse of the Wilson **B** matrix. These coordinates are not orthogonal. We construct the matrix whose columns are the nonorthogonal vectors,

$$\mathbf{S}_{\text{reduced}} = \begin{bmatrix} \mathbf{r}^{(1)} & \mathbf{r}^{(2)} & \cdots & \mathbf{r}^{(R)} \end{bmatrix}$$
(7.8)

The singular value decomposition of **S** gives you the form

$$\mathbf{S}_{\text{reduced}} = \mathbf{U}_{\text{reduced}} \mathbf{\Sigma}_{\text{reduced}} \mathbf{W}_{\text{reduced}}^{T}$$
(7.9)

and one can also compose the Grammian,  $\mathbf{G}_{reduced} = \mathbf{S}_{reduced}^T \mathbf{S}_{reduced}$ . The first *R* columns of the matrix **U** form a basis for the reduced space,  $\mathbf{V}_{reduced}$ . Another basis is obtained by Löwdin orthogonalization of  $\mathbf{S}$ ,<sup>6</sup>

$$\mathbf{V}_{\text{reduced}} = \mathbf{S}_{\text{reduced}}^{T} \mathbf{G}_{\text{reduced}}^{-1/2} = \mathbf{U}_{\text{reduced}} \operatorname{step}(\mathbf{\Sigma}_{\text{reduced}}) \mathbf{W}_{\text{reduced}}^{T}$$
(7.10)

This gives the orthogonal basis that is as close as possible to the original nonorthogonal basis;<sup>7-9</sup> Eq. (7.10) is equivalent to the procedure we used in chapter 3.  $\mathbf{G}_{reduced}^{-1/2}$  should be computed as a generalized inverse (in case the initial vectors are linearly dependent). In the formula based on singular-value decomposition, the same objective is achieved by defining

$$\operatorname{step}(\boldsymbol{\Sigma}_{\operatorname{reduced}}) = \begin{cases} 1 & \operatorname{elements of } \boldsymbol{\Sigma}_{\operatorname{reduced}} \text{ greater than } \boldsymbol{\varepsilon} \\ 0 & \operatorname{elements of } \boldsymbol{\Sigma}_{\operatorname{reduced}} \text{ less than or equal to } \boldsymbol{\varepsilon} \end{cases}$$
(7.11)

where  $\varepsilon > 0$  is a small cutoff to allow for roundoff error in the orthogonalization procedure.

After constructing the reduced space, we form the nonreduced space. To do this, we first form a set of  $3N_{\text{atoms}} - 6(5)$  orthogonal vectors, which we took from the singular value decomposition of the Wilson **B** matrix,

$$\mathbf{B} = \mathbf{U}_{\mathbf{B}} \boldsymbol{\Sigma}_{\mathbf{B}} \mathbf{W}^T \tag{7.12}$$

We then choose the nonsingular vectors of  $U_B$ ,

$$\mathbf{S}_{\text{nonreduced}} = \begin{bmatrix} \mathbf{u}^{(1)} & \mathbf{u}^{(2)} & \cdots & \mathbf{u}^{(3N_{\text{atoms}} - 6(5))} \end{bmatrix}$$
(7.13)

and project out the component that lies in the reduced space,

$$\tilde{\mathbf{S}}_{\text{nonreduced}} = \begin{bmatrix} \tilde{\mathbf{u}}^{(1)} & \tilde{\mathbf{u}}^{(2)} & \cdots & \tilde{\mathbf{u}}^{(3N_{\text{atoms}}-6(5))} \end{bmatrix}$$

$$= \mathbf{S}_{\text{nonreduced}} - \mathbf{V}_{\text{reduced}} \mathbf{V}_{\text{reduced}}^{T} \mathbf{S}_{\text{nonreduced}}$$
(7.14)

Taking the singular value decomposition of this matrix,

$$\tilde{\mathbf{S}}_{\text{nonreduced}} = \mathbf{U}_{\text{nonreduced}} \mathbf{\Sigma}_{\text{nonreduced}} \mathbf{W}_{\text{nonreduced}}^{T}, \qquad (7.15)$$

We can construct a basis for the reduced space from the nonsingular vectors of  $U_{nonreduced}$  or by Löwdin orthogonalization,

$$\mathbf{V}_{\text{nonreduced}} = \tilde{\mathbf{S}}_{\text{nonreduced}} \left[ \left( \tilde{\mathbf{S}}_{\text{nonreduced}} \right)^T \tilde{\mathbf{S}}_{\text{nonreduced}} \right]^{-1/2}$$
  
=  $\mathbf{U}_{\text{nonreduced}} \text{step}(\boldsymbol{\Sigma}_{\text{reduced}}) \mathbf{W}_{\text{nonreduced}}^T$  (7.16)

The vectors that are obtained from Eq. (7.16) are as close as possible to the original singular vectors of **B**, and therefore as close as possible to the delocalized internal coordinates of Baker.<sup>5</sup> We tested the singular value decomposition method and the orthogonalization method, and found that the Löwdin orthogonalization method worked much better.

Are there even better ways to define the delocalized internal coordinates? For example, we can make the Hessian as diagonal as possible by using the normal vibrational modes as the initial basis of 3N atoms – 6(5) vectors. (I.e., put the normal vibrational modes, obtained by diagonalizing the quasi-Newton Hessian, into Eq. (7.13).) Eq. (7.16) would then give the set of nonredundant coordinates that maximally resembled the normal vibrational modes. We have not tested this choice, but it may make it easier to approximate the quasi-Newton Hessian.

Alternatively, we could choose to localize the nonredundant coordinates. Define the position of a redundant internal coordinate, in Cartesian coordinates, as

$$\mathbf{x}_{k} = \frac{\sum_{\text{atoms } a \text{ in internal coordinate } \mathbf{q}_{k}}{m_{a} \mathbf{x}_{a}}$$
(7.17)

where the summations includes only the positions of the two, three, or four atoms that contribute to the interatomic distance, angle-bending, or torsional coordinate, respectively. The first two moments of the position of a delocalized nonreduced nonredundant coordinate are

$$\left\langle \mathbf{y}^{(i)} \right\rangle = \sum_{k=1}^{N_{\text{int}}} \left| v_k^{(i)} \right|^2 \mathbf{x}_k \qquad i = R+1, R+2, \dots 3N_{\text{atoms}} - 6(5) \qquad (7.18)$$

$$\left\langle \left( y^{(i)} \right)^2 \right\rangle = \sum_{k=1}^{N_{\text{int}}} \left| v_k^{(i)} \right|^2 \left| \mathbf{x}_k \right|^2 \qquad i = R+1, R+2, \dots 3N_{\text{atoms}} - 6(5) \qquad (7.19)$$

The Boys localization procedure, applied to the redundant internal coordinate, is equivalent to minimizing, over all unitary transformations of the delocalized nonreduced nonredundant coordinates,

$$\underbrace{\min_{\mathbf{U}}}_{\mathbf{U}} \sum_{i=R+1}^{3N_{\text{atoms}}-6(5)} \left( \left\langle \left( y^{(i)} \right)^2 \right\rangle - \left\langle \mathbf{y}^{(i)} \right\rangle \cdot \left\langle \mathbf{y}^{(i)} \right\rangle \right) \\
= \underbrace{\max_{\mathbf{U}}}_{\mathbf{U}} \sum_{i=R+1}^{3N_{\text{atoms}}-6(5)} \left\langle \mathbf{y}^{(i)} \right\rangle \cdot \left\langle \mathbf{y}^{(i)} \right\rangle$$
(7.20)

### 5. Redundant Alternatives to Delocalized Internal Coordinates

While it is convenient, it is not strictly necessary to use nonredundant coordinates in the transition-state optimization. Specifically, one can append the singular vectors (the columns of  $U_B$  with zero eigenvalue) to the v-space, as constructed with one of the methods from the previous section. This defines the orthogonal basis

$$\mathbf{W} = \begin{bmatrix} \mathbf{v}^{(1)} & \cdots & \mathbf{v}^{(3N_{\text{atoms}} - 6(5))} & \mathbf{u}_{\mathbf{B}}^{(3N_{\text{atoms}} - 7(6))} & \cdots & \mathbf{u}_{\mathbf{B}}^{(N_{\text{int}})} \end{bmatrix}$$
(7.21)

Using **W**, instead of **V**, means that the Hessian that is used in the transition-state optimization is larger ( $N_{int} \times N_{int}$  instead of  $3N_{atoms} - 6(5) \times 3N_{atoms} - 6(5)$ ). Because of this, the Hessian can store additional information, which we believed would speed the optimization. We found, however, that the quasi-Newton Hessian rarely had more than  $3N_{atoms} - 6(5)$  eigenvalues with significant magnitude. Using **W**, instead of **V**, in saddle caused small changes in the number of iterations required to converge specific reactions. On average, the **W**-based optimizer required one additional iteration to converge.

### 6. Finding the best Quasi-Newton Hessian

In Chapter 3, we noticed that the performance of transition-state optimizers is highly sensitive to the model that is used to update the Hessian. All methods for updating the Hessian use the difference in the gradients between the last two steps,

$$\mathbf{y}_{\text{conventional}} = \mathbf{g}_{\text{new}} - \mathbf{g}_{\text{old}} \tag{7.22}$$

to provide information about the second derivative in the direction of the step; this information is used to update the current approximation to the Hessian. However, in recent years, authors have examined alternative ways to define the **y**-vector which incorporate partial information about the effects of third derivatives;<sup>10-14</sup> we have implemented seven of these alternative **y**-vector definitions into *Saddle*.

Any of these seven alternative **y**-vector definitions can be used in place of the conventional definition, Eq. (7.22), in any quasi-Newton formula. We implemented thirteen different quasi-Newton methods,<sup>15-21</sup> all of which can be used either with or without damping, into *Saddle*.

To test these methods, we used the protocol from Chapter 4 on the Baker set of 25 reactions.<sup>22</sup> Specifically, we used Eq. (4.4) to obtain random displacements in the key internal coordinates such that the final structure was either  $\varepsilon = 0.1$  a.u. (five "good" guesses) or  $\varepsilon = 0.4$  a.u. (five "bad" guesses) away from the initial structure. To test the quasi-Newton methods for transition-state optimization, we considered deformations of the transition-state structure. To test the quasi-Newton methods for geometry minimization, we considered deformations of the reactant and product structures.

For minimization problems, the most promising optimization methods were the BFGS,<sup>23-27</sup> PSB,<sup>70,71</sup> Hoshino,<sup>28</sup> Xie,<sup>29</sup> and Bofill2003b<sup>15</sup> updates, together with either the conventional definition of the **y**-vector or the definitions from <sup>13</sup>, from Eq. (2.14) of <sup>10</sup>, or from Babaie-Kafaki <sup>14</sup>. For transition-state optimization, we found that the PSB, SR1, Bofill2003b, and Bofill's TS-BFGS formula methods worked best.<sup>17</sup> The best choices for the **y**-vector were the conventional definitions and the definitions from <sup>13</sup> or from Eq. (2.14) of <sup>10</sup>. We should extend our study to the full database of reactions to establish stronger preferences between these methods. We also need to investigate whether using damped-quasi-Newton methods improves the efficacy of these methods.

Among the best quasi-Newton methods for transition-states here are Bofill's hybrid methods, which mix two different quasi-Newton formulae.<sup>30,31</sup> We should consider combinations of quasi-Newton methods that are excellent for minimization with quasi-Newton methods that work well for transition-state optimization. As was done for the initial screening of quasi-Newton methods, these hybrid methods will be tested using 5 random "good" initial guesses and 5 random "bad" initial guesses. Preliminary results will be done for the 25 reactions, and then fewer combinations will be used for the whole 130 reactions.

### 7. Testing Robust Dihedral Coordinates

Chapter 2 presents a new way to represent dihedral angles, termed robust dihedral descriptors (cf. Eqs. (2.5) and (2.6)). These descriptors avoid the numerical problems that arise when one describes the dihedral rotation  $\alpha\beta\gamma\delta$  in the traditional way if one of the

angles  $\alpha\beta\gamma$  or  $\beta\gamma\delta$  is nearly linear. We used the robust dihedral descriptors, instead of the conventional dihedral angles, throughout the remainder of the thesis.

While the robust dihedral descriptors prevent one of the sources of numerical instability that can cause other programs to crash, we should test whether or not they compromise the performance of our method. (For most optimizations, certainly, there is no problem with the conventional approach.) We should also test whether our method for choosing which dihedral angles to include in the set of redundant internal coordinates works well compared to conventional approaches (e.g., the selection criterion used in the *Dalton* program<sup>32</sup>) or just choosing all dihedral angles, for every bond in the molecule.

To study this, we considered the two different methods for specifying dihedral rotations (our robust description and the conventional method) and three different ways to choose the dihedral coordinates (our method from Chapter 2, the method in *Dalton*, and including all dihedrals). We generated ten initial guess geometries for each of the reactions in our test set by randomly changing the key internal coordinates such that the final structure was either  $\varepsilon = 0.1$  a.u. (five "good" guesses) or  $\varepsilon = 0.4$  a.u. (five "bad" guesses) away from the initial structure. We observed that using the robust dihedral descriptors converges a larger percentage of the reactions, and that all the methods have roughly similar computational cost for reactions where they all converge.

### 8. **Optimizing Program Parameters**

As discussed at the end of Chapter 3, there are many user parameters in *Saddle* and we have not, thus far, made any attempt to systematically optimize them. Many of the choices can be validated separately, and the previous six subsections give insight into how this may be done. But, especially when considering parameters that control the stepsize and the Hessian update, it is probably important to optimize the parameters jointly.

There are two dichotomic choices (using rational function optimization (RFO) or trust region image method (TRIM) to control the step size; using the energy-based or gradient-based criteria for updating the trust radius) and nine to eleven numerical parameters ( $\omega$  and  $\upsilon$  from Eqs. (3.43)-(3.44), which determine when a finite-difference computation for a row/column of the Hessian is performed;  $\kappa$  in Eq. (3.55), which determines whether a quasi-Newton update should be performed in the reduced space;  $\lambda_p$ and  $\lambda_n$  from Eq. (3.57), which control the eigenstructure of the Hessian; the values of the maximum, minimum, and initial trust radii; and four to six parameters specifying the trust radius scheme (specifying whether a step is good and how much the trust radius should increase if so; specifying whether a step is poor and how much the trust radius should decrease if so).

We did not optimize the values of these parameters, or even search for good choices of these parameters by trial-and-error. This suggests that the method is not very sensitive to these parameters, within reason. We would like to select two sets of parameters: one that minimizes the failure rate of the algorithm when the initial guess is bad, and one that maximizes the speed of the algorithm when the initial guess is good. To

do this, we will need to first enhance our testing set: we would like to roughly the double the size of our testing database, to 250 reactions, paying special attention to reactiontypes (especially organometallic chemistry and transition-metal chemistry) that are underrepresented in the current database. We would also like to form a set of about 50 very small reactions, which can be studied at the post-Hartree-Fock level. Then, we can use ten "good guesses" (random displacements of the reduced coordinates of the transition state structure with magnitude  $\varepsilon = 0.1$  a.u.) and ten "bad guesses" ( $\varepsilon = 0.4$  a.u.) to choose optimal parameters for speed (for "good guesses") and robustness (for "bad guesses"). To optimize the parameters, we will use response surface methodology.<sup>33-38</sup>

### 9. Extensions: Minimization and Frozen Coordinates

The same method we developed for transition-state optimization can be extended to minimization. (The only essential change is that one requires the Hessian to have zero negative eigenvalues, instead of just one.) For minimization problems, it is also less obvious how to choose the reduced coordinates but, fortunately they are less-needed. In our initial testing, our algorithm is competitive with other quasi-Newton-based geometry minimization methods. More thorough testing will be presented later, together with the work on optimizing quasi-Newton methods.

It is often interesting to constrain the value of certain coordinates: this allows one to restrict conformational changes, and also to drive a reaction from its reactant to its product structure. We developed a method to freeze the value of one or more coordinates. The idea is quite similar to the method in Chapter 3. First one defines the constraints, requiring certain coordinates (which can be arbitrary linear combinations of redundant internal coordinates or atomic positions) to be frozen at specified values,

$$\phi^{(i)} = \mathbf{f}^{(i)} = \sum_{k=1}^{N_{\text{int}}} c_i \mathbf{q}_i \qquad i = 1, 2, \dots, F \qquad (7.23)$$

We then form an orthogonal basis for the frozen space using the method we used in chapter 3, section 3.3.F to define an orthogonal basis for the reduced coordinates. We then project out the components of the reduced coordinates corresponding to the frozen subspace, and form an orthogonal basis for the reduced coordinates using the method that was used to define the delocalized nonredundant coordinates in chapter 3, section 3.3.F. Finally, we project out the components of the delocalized internal coordinates corresponding to the frozen and reduced subspaces, and construct an orthogonal basis for this subspace. The resulting  $\mathbf{V}$  matrix, defining the orthogonal vectors in the  $\mathbf{v}$ -space, has the form

$$\mathbf{V} = \begin{bmatrix} \mathbf{v}^{(1)} & \cdots & \mathbf{v}^{(F)} & \mathbf{v}^{(F+1)} & \cdots & \mathbf{v}^{(F+R)} & \mathbf{v}^{(F+R+1)} & \cdots & \mathbf{v}^{(3N_{\text{atoms}}-6(5))} \end{bmatrix}$$
(7.24)

To prevent the optimization from making large steps in the v-space, we set the diagonal elements of frozen-coordinate block of the v-space Hessian to be large positive numbers. (This is equivalent to imposing a quadratic penalty constrain the values of these coordinates.) An optimization step in v-space is computed, and the components corresponding to the frozen subspace (the first F components of the step vector) are set equal to zero, so that the constraints are satisfied. Next, the Cartesian step is computed using the manifold projection method developed in chapter 2. The weight matrix in Eq. (2.7) is given very large components for the frozen coordinates, so that the Cartesian

coordinates that we use to evaluate the energy and gradient for the next step of the optimization will almost perfectly satisfy the constraints.

We have not systematically tested this method, but we have used it in several contexts, both for routine computations and for the pathfinding algorithm presented in the next section. So far, our approach seems to be more robust than the methods for constrained optimization in the *Gaussian* program.

# C. Topics Not Covered in the Thesis

#### **1.** Finding the Reaction Path with Sequential Quadratic Programming

This thesis focussed on determining the transition-state structure, which is (together with the more easily located reactant and product structures) enough to describe chemical kinetics and thermodynamics at the level of transition-state theory. However, sometimes one wishes to have not only the structures of the reactant, transition state, and product, but the entire reaction coordinate connecting them. This can be useful conceptually,<sup>39-42</sup> but it can also be useful practically, for evaluating corrections to traditional transition-state theory.<sup>43-48</sup>

Although there are many different ways to define the reaction coordinate, one usually uses the minimum energy pathway from the reactant to the product.<sup>49</sup> This pathway is the "leading line" about which reactive trajectories cluster.<sup>50</sup>

As part of my Ph.D. research, I developed a new sequential quadratic programming method (SQPM) for finding the minimum energy path.<sup>51</sup> At each step of the algorithm, one has an approximate reaction path, which is specified by a list of molecular

structures, represented in redundant internal coordinates,  $\{\mathbf{q}_i\}_{i=1}^{N_{\text{points}}}$ . Using a cubic spline, we define a continuous reaction coordinate,  $\mathbf{q}(t)$ , that passes through the points. The goal of the SQPM algorithm is to slide the entire pathway downhill, towards the true minimum energy reaction path.

For the true minimum-energy reaction path,  $\mathbf{q}_{\text{MEP}}(t)$ , the gradient of the potential energy at each point on the path is perpendicular to the direction of the path. That is,

$$\nabla U\left(\mathbf{q}_{\text{MEP}}\left(t\right)\right) \cdot \frac{d\mathbf{q}_{\text{MEP}}\left(t\right)}{dt} = 0$$
(7.25)

Since this condition holds for the optimal path, imposing at each stage of the optimization does not prevent the SQPM method from converging to the true minimum-energy pathway. Therefore the SQPM path is updated by minimizing the energy of all the intermediate points on the pathway on the hyperplane for which Eq. (7.25) holds. That is,

$$\mathbf{q}_{i}^{(\text{new})} = \arg \min_{\left\{\mathbf{q} \mid \frac{d\mathbf{q}_{i}^{(\text{old})}}{dt} \cdot (\mathbf{q} - \mathbf{q}_{i}^{(\text{old})}) = 0\right\}} U(\mathbf{q}_{i}) \qquad i = 2, 3, \dots, N_{\text{points}} - 1 \quad (7.26)$$

We do not assume that the endpoints of the path are already at local minima, and they are optimized without constraints. Each optimization in Eq. (7.26) is a geometry minimization, with a frozen coordinate by the constraint

$$\frac{d\mathbf{q}_{i}^{(\text{old})}}{dt} \cdot \mathbf{q} = \frac{d\mathbf{q}_{i}^{(\text{old})}}{dt} \cdot \mathbf{q}_{i}^{(\text{old})}$$
(7.27)

At each step of the SQPM procedure, we perform one step of this optimization using the method described in section 7.2.B.8, using a trust radius to ensure that the stepsize does

not move beyond the region where our quadratic model for the energy (constructed from the computed potential energy and its gradient, along with the approximated quasi-Newton Hessian) is reliable. Using the new points, we can construct a new reaction path and recompute the tangent vector to the path,  $d\mathbf{q}(t)/dt$ . We are then ready to perform the next step of the optimization using the same protocol (Eq. (7.26)). While the idea of SQPM is very simple, there are several nuances that complicate the implementation of the idea. First of all, after several steps of the procedure, one usually observes that the points have "bunched up" in low-energy regions of the potential energy surface. It is therefore necessary to redistribute the points along the pathway. We wish to ensure that there are more points in the most important chemical regions (near the reactant, transition-state and product structures, where the curvature of the potential energy surface is high); we also wish to ensure that there are more points in regions where it is difficult to accurately specify the reaction coordinate (where the curvature of the reaction path is high). Accordingly, at each stage of the algorithm, we redistribute the points on the path so that the points are not bunched together, and so that there are more points where the curvature of the energy, and/or the path itself, is high.

Second, there is a tendency for the path to develop a kink, which happens when the motion of the point creates a path that doubles-back on itself or crosses over itself. These kinks can be "cut out" when they are detected, but this slows the convergence of the method. We observed that if we relaxed the constraint in Eq. (7.27) by only including the key coordinates (by setting all the components of  $d\mathbf{q}(t)/dt$  that did not correspond to a user-specified coordinate to zero), most problems with kinking could be avoided.

Selecting the appropriate coordinates, unfortunately, sometimes requires significant insight.

We tested this method on a subset of twenty reactions from our transition-state database. In general, we were able to converge the path to the point where the magnitude of the projected gradient in Eq. (7.25) was less than 0.001 a.u. in around thirty iterations, which is significantly faster than the other approaches. (Finding the minimum energy path with very high accuracy however, is difficult using the SQPM method. Other methods are more suitable if higher accuracy than this is needed.) We verified that the transitionstate estimate from SQPM was truly excellent; from that starting point, transition-state optimizers converge almost immediately. Moreover, a good-enough guess for the transition state can be obtained using a looser convergence criterion than what we used in our systematic studies; investigating what convergence threshold is appropriate when using SQPM to provide initial guesses for transition-state optimization is a topic for future work. We also need to design better methods for dealing with the kinking problem.

As reviewed in the introductory chapter, there are alternative approaches for finding minimum energy pathways. I chose to work on SQPM because it seems to be especially suitable for reactions where it is challenging to find a good guess for the transition state. Other methods, like the quadratic string method,<sup>52,53</sup> are more appropriate when a highly accurate approximation to the minimum energy path is needed. Like string methods, SQPM also applies to complex, multi-step reactions with multiple reactive intermediates and transition states.

### 2. Study of the Epoxide Hydrolase Enzyme

Computational modeling of transition states is especially important for chemical reactions involving large molecules, like enzyme-catalyzed reactions. In these cases the experimental tools that can be used to elucidate the structure of the transition state are very limited. Often the only practical experiment is the measurement of the reaction rate. Merely measuring the rate of reaction does not provide much insight into the transition-state structure. However, if one performs an isotopic substitution on the system and then observes a shift in the reaction rate, that suggests that the atom for which the isotopic substitution was performed plays an important role in the reaction mechanism. Kinetic-isotope effects (KIEs) and equilibrium isotope effects (EIEs) therefore provide indirect information about the molecular structure of the reactant, transition state, and product.<sup>54-56</sup>

To use KIEs to learn the structure of a transition state, one compares the results of computed KIEs with experimentally measured KIEs. If the computed KIEs resemble the measured KIEs, this strongly suggests that the computed transition-state structure strongly resembles the true structure, validating one's calculation.<sup>57-60</sup> KIEs, therefore, are one of the few objective tools that one has for validating computational models for transition-state structures.

During my Ph.D. I worked to develop a program, *zebra*, for computing kinetic isotope effects not only for small molecules, but for large systems computed with quantum-mechanics/molecular-mechanics (QM/MM) hybrid methods.<sup>61-73</sup> In QM/MM calculations, the portion of the system (a solute and the nearest solvent molecules; a reaction center and nearby portions of a macromolecule) that is subject to bond-

breaking/bond-forming events is treated explicitly, and the remainder of the system is treated using classical molecular mechanics force fields. This allows one to treat chemical processes in complex environments, where direct quantum-mechanical modeling is impractical.

As an initial application of *zebra*, I computed KIEs for the epoxide hydrolase enzyme. Conventionally, KIEs in enzymes are computed by building a cluster model (that is, treating only a small fragment of the enzyme), finding the transition-state for the cluster model, and comparing the computed KIEs with the measured KIEs. This neglects the oft-important electrostatic interactions of the active site with its environment. To avoid this approximation, I used QM/MM calculations to compute the KIEs.

I started by performing a traditional QM/MM study of the enzyme mechanism. In particular, I studied the substrate specificity of epoxide hydrolase (EH) both for the human soluble EH (sEH)<sup>74</sup> and for mycobacterium tuberculosis (*Mtb*) epoxide hydrolase enzyme EHB<sup>75</sup> for three substrates: trans-1,3-diphenylpropene oxide, trans-stilbene oxide and cis-stilbene oxide. I studied the alkylation and the hydrolysis steps for all three substrates. For the first substrate, I also studied the regioselectivity of the nucleophilic attack of the catalytic aspartate residue on either the benzylic or the homo-benzylic carbon for trans-1,3-diphenylpropene oxide. (The other substrates have only one dominant reaction site.) In agreement with experiments, my computations showed that the hydrolysis of the ester intermediate is the rate limiting step of the reaction, with barriers of 13-15 kcal/mol.<sup>76-78</sup> I also used reactivity indicators from conceptual DFT to elucidate the order of reactivity of the different epoxides. The reactivity indicators, especially the

dual descriptor,<sup>79-81</sup> largely predicted the correct trend in reactivity. Finally the electrostatic contribution to the stabilization of the transition state for the hydrolysis step was studied, which revealed that the transition state is stabilized predominately by seven conserved amino acids between human sEH and EHB.

Next I computed the secondary tritium KIE for the alkylation step. To do this, I used the frequencies computed from the QM/MM calculation, with the QM fragment fixed in the electrostatic field of the MM atoms. The calculated value was 1.27, in agreement with the experimental value 1.30 (for soybean epoxide hydrolase)<sup>82</sup>.

In performing this calculation, I became aware of several issues that affect the computation of kinetic isotope effects. First, there are several different formulas for kinetic isotope effects in the literature,<sup>83</sup> differing mainly by whether the imaginary frequency mode at the transition state (i.e., motion along the reaction coordinate at the transition state) is treated as a vibrational<sup>83-86</sup> or a translational mode.<sup>87-89</sup> (In my previous work, I used the Eyring formulation, where it was treated as a translation.<sup>89</sup>) There is also the question of how one should treat tunnelling corrections; I have used the Wigner tunnelling correction, but the model of Miller would be a more accurate choice.<sup>90</sup> Furthermore, it seems that some of the formulas that are commonly used for kinetic isotope effects implicitly include tunnelling corrections;<sup>83</sup> there is the risk of double-counting tunnelling effects, then.

To address which formula is best, one should form a database of unimolecular reactions where the kinetic isotope effect has been accurately measured, and then attempt to reproduce those results with high-level *ab initio* calculations. We will focus on

reactions where tunnelling makes an important contribution to the kinetic isotope effect. (A good rule of thumb is that if the KIE is not bracketed by 1.0 and the equilibrium isotope effect, then tunneling and other corrections to transition-state theory may be important.<sup>91</sup>) By comparing the results of various theoretical approaches to experiment, we can select the best approach to KIE computations.

The second issue that affects QM/MM simulations is how one should treat the vibrations of the environment of the QM region. There is considerable controversy about the origins of enzymatic catalysis and, in particular, if vibrations of the enzyme environment could possibly play a role in catalysis.<sup>69,92-107</sup> (There is little debate that enzyme vibrations contribute to the overall rate. The question is whether they change the rate relative to the solution-phase reaction (in which case they qualify as catalytic effects) or not.) Including *all* the vibrations of the enzyme is impractical and unwise: the kinetic effects of the vibrational modes of the QM system are negligible beside those of the MM system. A pragmatic approach is to freeze the atoms in the enzyme environment (e.g., by assigning them infinite mass). This, however, does not allow for the possibility that vibrations of the enzyme environment may affect the relative rates of the isotopically differentiated reactions.

To address this issue, I started to work with the mobile block Hessian (MBH) approach to macromolecular dynamics.<sup>108-113</sup> The idea is to regard the protein environment as a collection of rigid blocks, which can move relative to each other. The internal vibrational motions associated with bond-stretching and angle-bending are therefore suppressed, and only the low-frequency modes associated with large-scale

motions of the protein environment are included. For this reason, computing KIEs for enzymatic catalysis with using MBH should be even more accurate than an exact computation (where the effects of noncatalytic bond stretches obscure the desired effect) or a frozen-environment computation (where the motion of the protein environment is neglected). By comparing the KIEs computed from MBH and frozen-environment calculations, we can learn how important protein vibrations are for enzymatic kinetics.

Right now, *zebra* is in a half-finished state. Completing it would provide a valuable tool to the growing community of scientists using KIEs (and EIEs, which are easier) to elucidate the chemical reaction mechanisms. To validate *zebra*, it would be wise to first investigate enzymes where very good KIE data is available to us from Prof. Paul Berti's group (e.g., cAMP<sup>114</sup> and mutY.<sup>115</sup>). Next it would be interesting to study enzymes like where the reaction is coupled to a large-scale protein motion, like lysozyme and cephalexin (CEX). Finally, we could address enzymes with possible promoting vibrations (e.g., alcohol dehydrogenase).

#### **3.** Computational & Theoretical Studies of Chemical Reactivity

Can removing an electron from a molecule cause the electron density to increase in certain regions of the molecule? Somewhat surprisingly, this effect is nearly ubiquitous: a global decrease in electron number is almost always associated with local increases in electron density.<sup>116,117</sup> I collaborated on two papers where molecules that exhibit this effect were studied: in the first paper the focus was on molecules where there are multiple reactive frontier orbitals;<sup>2</sup> in the second paper the focus was on molecules with very small band gaps.<sup>3</sup> (Most of the molecules in the second paper were developed for use in molecular electronics.) These families of molecules were selected because they were expected to show dramatic local increases in electron density upon electron removal; this was observed. The effect was not as strong as we had hoped, however: we were hoping to find a case where removing an electron from a molecule caused the oxidation state of one of the atoms in the molecule to decrease. In these molecules, however, the increases in electron density were too delocalized to see this effect.

To extend this treatment to cases where one removes (or adds) an electron with a specified spin to a molecule, I studied how the energy and other properties of molecules respond to changes in the number of electrons with a given spin. The results of that inquiry are published in the appendix of ref.<sup>1</sup>. In unpublished work, I developed the idea from that paper more thoroughly and investigated the spin-dependent electron-transfer preferences of electrophiles and nucleophiles.

# 7.3 References:

(1) Cuevas-Saavedra, R.; Chakraborty, D.; Rabi, S.; Cardenas, C.; Ayers, P.

W. J. Chem. Theory Comp. 2012, 8, 4081.

(2) Echegaray, E.; Cardenas, C.; Rabi, S.; Rabi, N.; Lee, S.; Heidar Zadeh, F.;

Toro-Labbe, A.; Anderson, J. S. M.; Ayers, P. W. J. Mol. Model. 2013, 19, 2779.

(3) E. Echegaray, S. Rabi, C. Cárdenas, F. Heidar Zadeh, N. Rabi, S. Lee, J. S.

M. Anderson, A. Toro-Labbe, and P. W. Ayers. J. Mol. Modeling (accepted).

- (4) S. Rabi, S. K. Burger, P. W. Ayers, J. Mol. Modeling (submitted).
- (5) Baker, J.; Kessi, A.; Delley, B. J. Chem. Phys. 1996, 105, 192.
- (6) Löwdin, P. O. Advances in Physics **1956**, *5*, 1.
- (7) Aiken, J. G.; Erdos, J. A.; Goldstein, J. A. Int. J. Quantum Chem. 1980,

18, 1101.

- (8) Mayer, I. Int. J. Quantum Chem. 2002, 90, 63.
- (9) Carlson, B. C.; Keller, J. M. Phys. Rev. 1957, 105, 102.
- (10) Wei, Z. X.; Li, G. Y.; Qi, L. Q. Applied Mathematics and Computation **2006**, *175*, 1156.

(11) Zhang, J. Z.; Xu, C. X. *Journal of Computational and Applied Mathematics* **2001**, *137*, 269.

(12) Biglari, F.; Abu Hassan, M.; Leong, W. J. *Journal of Computational and Applied Mathematics* **2011**, *235*, 2412.

(13) Xiao, Y. H.; Wei, Z. X.; Wang, Z. G. Computers & Mathematics with Applications 2008, 56, 1001.

- (14) Babaie-Kafaki, S. Science China-Mathematics **2011**, *54*, 2019.
- (15) Bofill, J. M. Int. J. Quantum Chem. 2003, 94, 324.
- (16) Nocedal, J.; Wright, S. J. *Numerical Optimization*; Springer-Verlag: New

York, 1999.

- (17) Anglada, J. M.; Bofill, J. M. J. Comput. Chem. 1998, 19, 349.
- (18) Yuan, Y. X. Ima Journal of Numerical Analysis 1991, 11, 325.
- (19) Yuan, Y. X.; Byrd, R. H. Journal of Computational Mathematics 1995, 13,

95.

(20) Bungay, S. D.; Poirier, R. A. J. Mol. Struct.: THEOCHEM 2002, 591, 35.

(21) Yuan, Y. X. In Wiley Encyclopedia of Operations Research and

Management Science; Cohran, J. J., Cox, L. A., Keskinocak, P., Kharoufeh, J. P., Smith,

J. C., Eds.; Wiley: 2010.

(22) Baker, J.; Chan, F. R. J. Comput. Chem. 1996, 17, 888.

(23) Broyden, C. G. Journal of the Institute of Mathematics and Its

Applications 1970, 6, 76.

- (24) Fletcher, R. Computer Journal **1970**, 13, 317.
- (25) Goldfarb, D. *Mathematics of Computation* **1970**, *24*, 23.
- (26) Shanno, D. F. *Mathematics of Computation* **1970**, *24*, 647.
- (27) Shanno, D. F.; Kettler, P. C. Mathematics of Computation 1970, 24, 657.

(28) Hoshino, S. Journal of the Institute of Mathematics and Its Applications**1972**, 10, 394.

(29) Xie, Y. F. Acta Math. **1989**, *32*, 721.

- (30) Bofill, J. M. J. Comput. Chem. 1994, 15, 1.
- (31) Bofill, J. M. Chem. Phys. Lett. 1996, 260, 359.
- (32) Bakken, V.; Helgaker, T. J. Chem. Phys. 2002, 117, 9160.
- (33) Robinson, T. J.; Borror, C. M.; Myers, R. H. Quality and Reliability

Engineering International 2004, 20, 81.

(34) Chen, V. C. P.; Tsui, K. L.; Barton, R. R.; Meckesheimer, M. *Iie Transactions* **2006**, *38*, 273.

(35) Myers, R. H.; Montgomery, D. C.; Vining, G. G.; Borror, C. M.;

Kowalski, S. M. Journal of Quality Technology 2004, 36, 53.

- (36) Myers, R. H.; Khuri, A. I.; Carter, W. H. *Technometrics* **1989**, *31*, 137.
- (37) Bezerra, M. A.; Santelli, R. E.; Oliveira, E. P.; Villar, L. S.; Escaleira, L.

A. Talanta 2008, 76, 965.

(38) Simpson, T. W.; Peplinski, J. D.; Koch, P. N.; Allen, J. K. *Engineering* with Computers **2001**, *17*, 129.

- (39) Fukui, K. Acc. Chem. Res. 1981, 14, 363.
- (40) Toro-Labbe, A. J. Phys. Chem. A **1999**, 103, 4398.
- (41) Toro-Labbe, A.; Gutierrez-Oliva, S.; Politzer, P.; Murray, J. S. In

*Chemical Reactivity Theory: A density functional view*; Chattaraj, P. K., Ed.; CRC Press: Boca Raton, 2009, p 293.

(42) Guevara-Garcia, A.; Echegaray, E.; Toro-Labbe, A.; Jenkins, S.; Kirk, S.

R.; Ayers, P. W. J. Chem. Phys. 2011, 134, 234106.

(43) Hynes, J. T. Annu. Rev. Phys. Chem. 1985, 36, 573.

- (44) Dumont, R. S.; Brumer, P. J. Phys. Chem. 1986, 90, 3509.
- (45) Carpenter, B. K. In Annu. Rev. Phys. Chem. 2005; Vol. 56, p 57.
- (46) Carpenter, B. K. *Nature Chemistry* **2010**, *2*, 80.
- (47) Carpenter, B. K. Science **2011**, *332*, 1269.
- (48) Collins, P.; Carpenter, B. K.; Ezra, G. S.; Wiggins, S. J. Chem. Phys.

**2013**, *139*.

- (49) Quapp, W.; Heidrich, D. Theor. Chim. Act. 1984, 66, 245.
- (50) Gonzalez, J.; Gimenez, X.; Bofill, J. M. PCCP 2002, 2921.
- (51) Burger, S. K.; Yang, W. J. Chem. Phys. 2007, 127, 164107.
- (52) E, W. N.; Ren, W. Q.; Vanden-Eijnden, E. Phys. Rev. B 2002, 66, 052301.
- (53) Burger, S. K.; Yang, W. T. J. Chem. Phys. 2006, 124, 054109.
- (54) Isotope effects in chemistry and biology; Kohen, A.; Limbach, H.-H., Eds.;

CRC: Boca Raton, 2006.

- (55) Bigeleisen, J.; Wolfsberg, M. Adv. Chem. Phys. 1958, 1, 15.
- (56) Liu, Q.; Tossell, J. A.; Liu, Y. Geochim. Cosmochim. Acta 2010, 74, 6965.
- (57) Schramm, V. L.; Horenstein, B. A.; Bagdassarian, C. K.; Schwartz, S. D.;

Berti, P. J.; Rising, K. A.; Scheuring, J.; Kline, P. C.; Parkin, D. W.; Merkler, D. J. Int. J.

Quantum Chem. 1996, 60, 81.

- (58) Schramm, V. L. Curr. Opin. Chem. Biol. 2001, 5, 556.
- (59) Schramm, V. L. Acc. Chem. Res. 2003, 36, 588.
- (60) Berti, P. J.; McCann, J. A. B. Chem. Rev. 2006, 106, 506.
- (61) Warshel, A.; Levitt, M. J.Mol Bio. 1976, 103, 226.

(62) Warshel, A. Proc. Natl. Acad. Sci. 1978, 75 5250.

(63) Hwang, J. K.; King, G.; Creighton, S.; Warshel, A. J. Am. Chem. Soc.

**1988**, 110, 5297.

- (64) Trajbl, M.; Hong, G. Y.; Warshel, A. J. Phys. Chem. B 2002, 106, 13333.
- (65) Elber, R.; Karplus, M. Chem. Phys. Lett. 1987, 139, 375.
- (66) Field, M. J.; Bash, P. A.; Karplus, M. J. Comput. Chem. 1990, 11, 700.
- (67) Gogonea, V.; Suarez, D.; Van Der Vaart, A.; Merz, K. W. Current

Opinion in Structural Biology 2001, 11, 217.

(68) Field, M. J. J. Comput. Chem. 2002, 23, 48.

(69) Shurki, A.; Warshel, A. In *PROTEIN SIMULATIONS Advances in protein chemistry* 2003, p 249.

- (70) Hu, H.; Yang, W. T. J. Mol. Struct.: THEOCHEM 2009, 898, 17.
- (71) Senn, H. M.; Thiel, W. Angewandte Chemie-International Edition 2009,

48, 1198.

(72) Zhang, Y.; Liu, H.; Yang, W. J. Chem. Phys. 2000, 112, 3483.

(73) Cisneros, G. A.; Liu, H. Y.; Zhang, Y. K.; Yang, W. T. J. Am. Chem. Soc.2003, 125, 10384.

(74) Mladenovic, M.; Fink, R. F.; Thiel, W.; Schirmeister, T.; Engels, B. J. Am. Chem. Soc. 2008, 130, 8696.

(75) Biswal, B. K.; Morisseau, C.; Garen, G.; Cherney, M. M.; Garen, C.; Niu,C. Y.; Hammock, B. D.; James, M. N. G. *J. Mol. Biol.* 2008, *381*, 897.

(76) Lacourciere, G. M.; Armstrong, R. N. J. Am. Chem. Soc. 1993, 115, 10466.

(77) Lacourciere, G. M.; Armstrong, R. N. Chem. Res. Toxicol. 1994, 7, 121.

(78) Rink, R.; Fennema, M.; Smids, M.; Dehmel, U.; Janssen, D. B. J. Biol.

*Chem.* **1997**, *272*, 14650.

- (79) Fuentealba, P.; Parr, R. G. J. Chem. Phys. 1991, 94, 5559.
- (80) Morell, C.; Grand, A.; Toro-Labbé, A. J. Phys. Chem. A 2005, 109, 205.
- (81) Ayers, P. W.; Morell, C.; De Proft, F.; Geerlings, P. Chem, Eur. J 2007,

13, 8240.

(82) Blee, E.; Summerer, S.; Flenet, M.; Rogniaux, H.; Van Dorsselaer, A.;

Schuber, F. J. Biol. Chem. 2005, 280, 6479.

- (83) Bigeleisen, J.; Wolfsberg, M. J. Chem. Phys. 1953, 21, 1972.
- (84) Bigeleisen, J. J. Phys. Chem. 1952, 56, 823.
- (85) Bigeleisen, J. J. Chem. Phys. 1949, 17, 675.
- (86) Bigeleisen, J.; Mayer, M. G. J. Chem. Phys. 1947, 15, 261.
- (87) McQuarrie, D. A.; Harper-Collins: New York, 1976.
- (88) Slater, N. B. Proc. R. Soc. London, Ser. A 1948, 194, 112.
- (89) Eyring, H. J. Chem. Phys. 1935, 3, 107.
- (90) Miller, W. H. J. Chem. Phys. 1973, 61, 1823.
- (91) Nigel Richards, personal correspondence.
- (92) Antoniou, D.; Schwartz, S. D. J. Phys. Chem. B 2001, 105, 5553.
- (93) Caratzoulas, S.; Schwartz, S. D. J. Chem. Phys. 2001, 114, 2910.
(94) Antoniou, D.; Caratzoulas, S.; Kalyanaraman, C.; Mincer, J. S.; Schwartz,S. D. *Eur. J. Biochem.* 2002, 269, 3103.

(95) Caratzoulas, S.; Mincer, J. S.; Schwartz, S. D. J. Am. Chem. Soc. 2002,

124, 3270.

- (96) Cui, Q. A.; Karplus, M. J. Phys. Chem. B 2002, 106, 7927.
- (97) Mincer, J. S.; Schwartz, S. D. Journal of Proteome Research 2003, 2, 437.
- (98) Mincer, J. S.; Schwartz, S. D. J. Phys. Chem. B 2003, 107, 366.
- (99) Hammes-Schiffer, S. Acc. Chem. Res. 2001, 34, 273.
- (100) Hammes-Schiffer, S. *Biochemistry* **2002**, *41*, 13335.
- (101) Benkovic, S. J.; Hammes-Schiffer, S. Science 2003, 301, 1196.
- (102) Villa, J.; Warshel, A. J. Phys. Chem. B 2001, 105, 7887.
- (103) Warshel, A.; Parson, W. W. Quarterly Reviews of Biophysics 2001, 34,

563.

- (104) Warshel, A.; Florian, J.; Strajbl, M.; Villa, J. Chembiochem 2001, 2, 109.
- (105) Kollman, P. A.; Kuhn, B.; Perakyla, M. J. Phys. Chem. B 2002, 106, 1537.
- (106) Shurki, A.; Strajbl, M.; Villa, J.; Warshel, A. J. Am. Chem. Soc. 2002, 124,

4097.

- (107) Warshel, A. Acc. Chem. Res. 2002, 35, 385.
- (108) Ghysels, A.; Van Neck, D.; Waroquier, M. J. Chem. Phys. 2007, 127.
- (109) Ghysels, A.; Van Neck, D.; Van Speybroeck, V.; Verstraelen, T.;

Waroquier, M. J. Chem. Phys. 2007, 126, 224102.

Ph.D. Thesis – Sandra Rabi; McMaster University – Chemistry and Chemical Biology

(110) Woodcock, H. L.; Zheng, W. J.; Ghysels, A.; Shao, Y. H.; Kong, J.;

Brooks, B. R. J. Chem. Phys. 2008, 129.

(111) Ghysels, A.; Van Speybroeck, V.; Verstraelen, T.; Van Neck, D.;

Waroquier, M. J. Chem. Theory Comp. 2008, 4, 614.

(112) Ghysels, A.; Verstraelen, T.; Hemelsoet, K.; Waroquier, M.; Van Speybroeck, V. J. Chem Inf. Model. 2010, 50, 1736.

(113) Ghysels, A.; Van Neck, D.; Brooks, B. R.; Van Speybroeck, V.;

Waroquier, M. J. Chem. Phys. 2009, 130.

- (114) McCann, J. A. B.; Berti, P. J. J. Am. Chem. Soc. 2007, 129, 7055.
- (115) McCann, J. A. B.; Berti, P. J. J. Am. Chem. Soc. 2008, 130, 5789.
- (116) Ayers, P. W.; Morrison, R. C.; Roy, R. K. J. Chem. Phys. 2002, 116, 8731.
- (117) Melin, J.; Ayers, P. W.; Ortiz, J. V. J. Phys. Chem. A 2007, 111, 10017.